

UNIVERSIDADE DE LISBOA

Faculdade de Ciências

Departamento de Informática



AVALIAÇÃO DO DESEMPENHO EM
CONTEXTO ORGANIZACIONAL UTILIZANDO
O MÉTODO BALANCED SCORECARD

Marco Nascimento Barros

MESTRADO EM ENGENHARIA INFORMÁTICA

Engenharia de Software

2013

UNIVERSIDADE DE LISBOA

Faculdade de Ciências

Departamento de Informática



AVALIAÇÃO DO DESEMPENHO EM
CONTEXTO ORGANIZACIONAL UTILIZANDO
O MÉTODO BALANCED SCORECARD

Marco Nascimento Barros

ESTÁGIO

Trabalho orientado pelo Prof. Doutor Dimitrios Mostrous e coorientado por Pedro
Manuel da Silva Roque e Almeida

MESTRADO EM ENGENHARIA INFORMÁTICA

Engenharia de Software

2013

Agradecimentos

Em primeiro lugar, quero agradecer à Contactus S.A. por ter apostado em mim, no âmbito do projeto de estágio e por me ter proporcionado grandes e bons momentos durante a realização do projeto. Um agradecimento especial ao Ademar Fernandes, à Ana Rosa, à Carla Silva, ao Filipe Alves, ao Pedro Almeida e ao Rui Guerreiro por todo apoio, compreensão e preocupação que tiveram comigo, tendo criado grandes amizades nesta empresa.

Em segundo lugar, agradeço a todos os meus amigos, especialmente à Ana Ganilho, ao André Godinho, ao António Pacheco, ao André Silva, ao Edgar Pólvora, ao Gonçalo Napoleão, à Inês Luís, ao Norberto Soares, ao Rui Paiva e ao Ricardo Wilhelm pela amizade, ajuda e pelos momentos de lazer que me proporcionaram no decorrer da Licenciatura e Mestrado.

Quero agradecer também a duas pessoas muito especiais na minha vida, que estiveram sempre presentes e que me reconheceram pelo meu valor. Um muito obrigado por tudo Beatriz Oliveira Tavares e Tatiana Cabral Mangerição.

Por fim, e mais importante, agradeço do fundo do meu coração aos meus pais por, dentro do possível, me terem dado sempre tudo mesmo nos momentos mais difíceis da nossas vidas. Pela paciência (que é bem precisa!) e dedicação que sempre tiveram para comigo, agradeço a Deus por ter os melhores e especiais pais do mundo.

Aos meus pais, o meu muito obrigado.

Resumo

O presente documento tem como objetivo detalhar o trabalho desenvolvido no estágio, de nove meses, realizado numa empresa ligada às tecnologias de informação, a Contactus S.A., no âmbito do Projeto em Engenharia Informática (PEI), integrado no 2º ano do Mestrado em Engenharia Informática da Faculdade de Ciências da Universidade de Lisboa (FCUL).

Com a crescente necessidade dos utilizadores de informação estruturada, em diferentes áreas, observa-se um maior investimento por parte das organizações na aquisição e implementação de Aplicações Web tornando-se estas peças vitais no planeamento e operação de uma organização. Este tipo de sistemas de informação é altamente personalizável, proporcionando uma integração robusta entre as diversas áreas de negócio de uma organização. Para estas instituições é a solução ideal para organizar todo o tipo de informação, deixando de utilizar em grande escala as quantidades gigantescas de informação em papel.

Tendo em conta que muitos Organismos têm inúmeras dificuldades em gerir os planos de atividades que estabelecem para cada ano, devido a determinados fatores, como por exemplo, alteração de objetivos inicialmente planeados, metas não cumpridas em determinadas datas, este projeto tende a ser a solução para uma melhor ajuda aos utilizadores para gestão de um plano de atividades.

Este trabalho tem pois como objetivo, a criação de um sistema de informação para os Organismos da Administração Pública, no sentido de dar suporte informático à elaboração do Plano de Atividade, um dos instrumentos fundamentais de gestão dos Organismos e uma necessidade cada vez mais sentida por estes, para avaliações objetivas dos serviços, baseadas em indicadores de medição quantitativa. Este sistema tem como nome, “strategiaGOV”, onde será implementado o módulo de suporte ao Plano de Atividades que recorre à metodologia Balanced Scorecard.

Palavras-chave: Balanced Scorecard, sistema de informação, plano de atividades, avaliações, objetivos.

Abstract

The current report has as its objective the detailing of the work developed in the on stage, nine months, in a company connected to the new information technologies, Contactus S.A., scope of the Project in Computer Engineering (PCE), integrated in the 2nd year of the Master's Degree in Computer Engineering at Faculdade de Ciências da Universidade de Lisboa (FCUL).

Along with (and due to) the growing demand from users of information structured in different areas, there is a noticeable increase in investment, acquisition and implementation of Web Applications from the part of corporations, thus making Web Applications a vital element in the design and operation of a company. This kind of information systems is highly customizable and grants involvement between the various business areas of a company. For these institutions, it is an ideal solution in what comes to organizing information of any sort, making use of large-scale massive amounts of information on paper.

Given that many Organisms have uncountable difficulties in managing yearly established activity schedules, due to diverse factors such as the revision of certain goals or unmet deadlines, this project presents itself as the key to better help users organize their plans.

This work has as aims to create of an information system for the Organisms of Administration Public, to provide computer support for the elaboration Activity Plan, a fundamental tools for management bodies. A need, ever more felt by them, for objective evaluations of services and employees, based on quantitative measurement indicators. The name of this system is "strategiaGOV" and will implement the support module from the activity plan, using the Balanced Scorecard methodology.

Keywords: Activity Plan, assessments, Balanced Scorecard, goals, information system.

Conteúdo

Lista de Figuras.....	vii
Lista de Abreviaturas	ix
Capítulo 1.....	1
Introdução	1
1.1 Motivação	1
1.2 Objetivos.....	3
1.3 Empresa de acolhimento.....	4
1.4 Integração profissional	5
1.5 Estrutura do documento.....	6
Capítulo 2.....	7
Metodologia e Planeamento.....	7
2.1 Metodologia de desenvolvimento.....	7
2.2 Planeamento do projeto	9
Capítulo 3.....	11
Análise e Desenho.....	11
3.1 Considerações gerais	11
3.1.1 Balanced Scorecard (BSC).....	12
3.1.2 Quadro de Avaliação e Responsabilização (QUAR)	14
3.2 Levantamento de Requisitos.....	15
3.3 Arquitetura da aplicação	18
3.3.1 Back End	19
3.3.2 Front End.....	20
3.3.3 Base de Dados	20
Capítulo 4.....	21
Contexto e enquadramento tecnológico	21
4.1 Ferramentas de desenvolvimento	21
4.2 Tecnologias utilizadas	23

4.2.1	jqGrid	23
4.2.2	Javascript e jQuery	24
4.2.3	PHP.....	25
4.2.4	PHPExcel	26
4.2.5	Oracle	27
Capítulo 5.....	28	
Trabalho desenvolvido.....	28	
5.1	Modelo de Dados.....	30
5.1.1	Administração do Sistema.....	30
5.1.1	Plano de Atividades.....	33
5.2	Implementação.....	36
5.2.1	Administração (Menu Admin)	38
5.2.2	Estrutura Orgânica.....	40
5.2.3	Planeamento	44
5.3	Gestão de Recursos.....	47
5.3.1	Ajuda	47
Capítulo 6.....	50	
Conclusão.....	50	
6.1	Trabalho realizado	50
6.2	Trabalho Futuro	51
6.3	Comentário Crítico	51
Bibliografia e Referências	54	

Lista de Figuras

Figura 1 – Enquadramento do Plano de Atividades no strategiaGOV	2
Figura 2 – Modelo Organizacional da Contactus (Fonte: Contactus S.A.).....	4
Figura 3 – Modelo em Cascata	7
Figura 4 – Modelo em Cascata Revisto	8
Figura 5 – Calendarização.....	10
Figura 6 – Hierarquia da Estrutura Orgânica	11
Figura 7 - Hierarquia das Perspetivas do BSC (Fonte: Wikipédia)	13
Figura 8 – Hierarquia dos Objetivos de um Organismo	15
Figura 9 – Arquitetura da aplicação strategiaGOV	18
Figura 10 – Modelo MVC (Model View Controller)	19
Figura 11 – Gestão de Projetos constituído por Subgrids	24
Figura 12 – Arquitetura da Framework jQuery (Fonte: J2eebrain)	25
Figura 13 – Funcionalidades da Aplicação	28
Figura 14 – Definição (Grupo 1)	29
Figura 15 - Controlo (Grupo 2).....	29
Figura 16 - Medição (Grupo 3).....	29
Figura 18 – Modelo de dados relativo à administração do sistema	31
Figura 17 – Listagem de Histórico relativo aos Ministérios.....	32
Figura 19 - Modelo de dados relativo ao Plano de Atividades	34
Figura 20 – Menu e ThemeRoller do strategiaGOV	37
Figura 21 – Tabela dos utilizadores	38
Figura 22 – Painel para administração dos perfis	39
Figura 23 – Processo de Registo da Estrutura Orgânica.....	40
Figura 24 – Gestão dos Ministérios	40
Figura 25 – Hierarquia Organizacional do Ministério	41

Figura 26 – Processo de definição dos Objetivos Estratégicos e Operacionais (BSC + QUAR)	42
Figura 27 – Hierarquia dos objetivos estratégicos e operacionais	43
Figura 28 – Gráfico gerado para Indicadores de um determinado objetivo.....	44
Figura 29 - Processo do Planeamento de Atividades	45
Figura 30 – Resumo das Atividades Correntes de um Organismo	45
Figura 31 - Processo de Registo de Iniciativas e Projetos	46
Figura 32 – Resumo dos projetos de um Organismo	46
Figura 33 – Processo de Planeamento dos Postos de Trabalho (RH)	47
Figura 34 - Processo de Planeamento da Formação Profissional	47
Figura 35 – Visão geral da construção do Plano de Atividades do Organismo.....	48

Lista de Abreviaturas

AJAX – Asynchronous Javascript and XML
API – Application Programming Interface
ASP – Active Server Pages
CSS – Cascading Style Sheets
CSV – Comma-Separated Values
DOM – Document Object Model
HTML – HyperText Markup Language
HTTP – Hypertext Transfer Protocol
IP – Internet Protocol
IDE – Integrated Development Environment
JSP – JavaServer Pages
JSON – JavaScript Object Notation
PDF – Portable Document Format
PHP – Hypertext Preprocessor
QUAR – Quadro de Avaliação e Responsabilização
SQL – Structured Query Language
TOAD – Tool for Oracle Application Developers
XML – eXtensible Markup Language

Capítulo 1

Introdução

Este relatório apresenta e descreve o projeto realizado como parte integrante da unidade curricular do Projeto de Engenharia Informática, leccionada na Faculdade de Ciências da Universidade de Lisboa, disciplina do 2º ciclo do Mestrado em Engenharia Informática no ano letivo 2012/2013, tendo a duração de nove meses.

Este projeto foi desenvolvido na empresa Contactus S.A. [1] , tendo paralelamente como objetivo a consolidação de conhecimentos na área de SI (Sistemas de Informação) e aquisição de competências na criação de sistemas de informação baseados em tecnologia *web*.

1.1 Motivação

As tecnologias têm vindo a evoluir de forma a suportar as necessidades duma sociedade cada vez mais tecnológica. Se observarmos bem, cada vez mais utilizamos sistemas de informação para nos ajudar nas atividades do nosso dia-a-dia, por exemplo, se precisarmos de consultar uma determinada informação, através de um sistema informático, conseguimos obtê-la num curto espaço de “*clicks*”, o que poderia levar horas se esta procura fosse realizada manualmente em arquivos em papel dispersos pelas várias instituições existentes. Tendo em conta esta necessidade por parte da sociedade têm sido desenvolvidas, em grande número, tecnologias que nos possibilitam a criação destes sistemas de informação que, por sua vez, potenciam a evolução tecnológica. É neste contexto que este trabalho se insere.

Este projeto tem como foco principal a criação de uma aplicação web para melhoria e automatização dos vários processos inerentes para a elaboração de um plano de atividades, tendo em conta que é necessário prever várias interligações com a gestão de projetos, com a avaliação de desempenho e com o orçamento de cada Organismo.

Para gerir um plano de atividades é essencial gerir toda a componente estratégica de uma instituição (figura 1), com isto, irá ser implementado o módulo de suporte ao Plano de Atividades que recorre à metodologia Balanced Scorecard [2], que é o foco principal de todo o projeto.

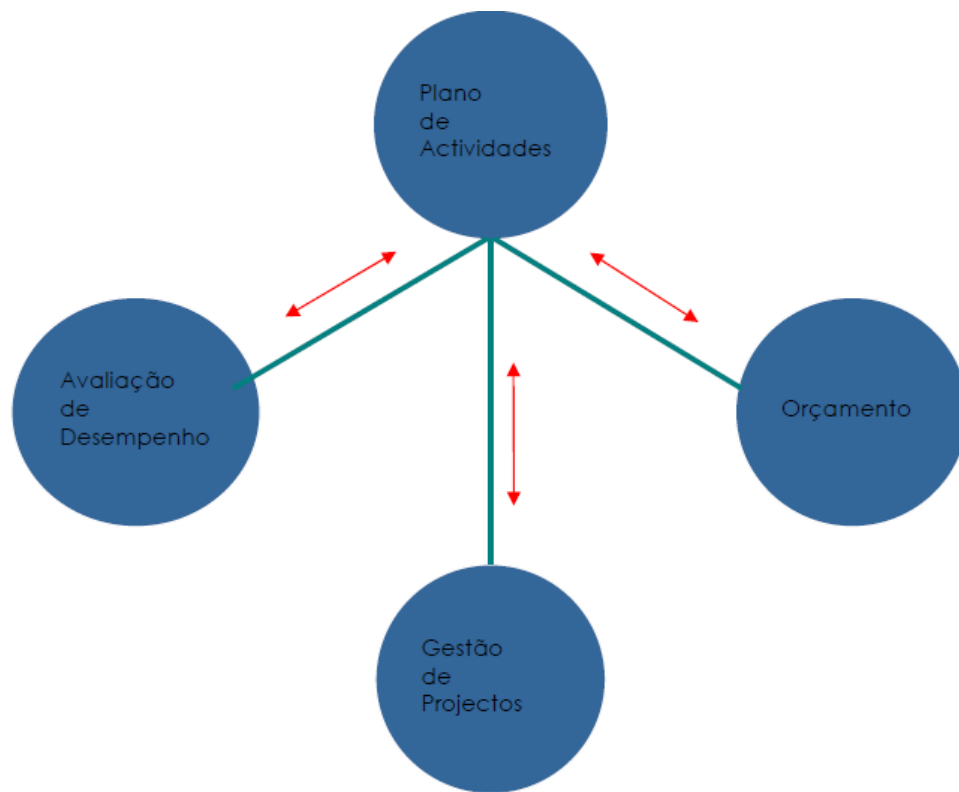


Figura 1 – Enquadramento do Plano de Atividades no strategiaGOV

O facto de se optar por uma aplicação baseada numa arquitetura *web* auxilia os utilizadores, visto que tudo o que necessitam é de um *browser* compatível com a aplicação e os seus componentes, sendo a maior parte do processamento realizado no lado do servidor. Com isto, é possível mostrar e colocar em prática todos os conhecimentos adquiridos ao longo de toda a formação académica e obter novos conhecimentos perante as várias tecnologias utilizadas neste projeto.

1.2 Objetivos

Este projeto, tem como principal objetivo, o desenvolvimento de uma aplicação *web*, no sentido de dar suporte informático à elaboração do Plano de Atividades, construindo todas as suas funcionalidades e respetivos processos.

Os principais aspetos a ter em consideração no âmbito do desenvolvimento do projeto são:

- ✓ Registrar toda a informação principal relativa aos Organismos, isto é, definir o seu enquadramento, a sua origem, história, missão, competências, visão e valores, etc;
- ✓ Definição dos objetivos do Organismo, tendo em conta a metodologia *Balanced Scorecard*;
- ✓ Desenvolver mecanismos que permitam registar projetos a serem realizados, apontando os recursos afetados (humanos e financeiros), prazos previstos, objetivos, indicadores e metas;
- ✓ Criar mecanismos que permitam uma relação de efetividade perante toda a informação registada;
- ✓ Desenvolver funcionalidades para registar as atividades correntes das Unidades Orgânicas, disponibilizando toda informação em quadro de resumo;
- ✓ Elaboração, e controle de execução, do Plano de Atividades e respetiva ligação ao Orçamento.

Adicionalmente, o PEI permitiu-me, assim, o enquadramento no mundo do trabalho, onde adquiri imensos conhecimentos, assimilando e desenvolvendo hábitos de trabalho inerentes à experiência profissional que se desenvolveu ao longo de todo o estágio na empresa, fortalecendo deste modo o meu crescimento a nível pessoal, social e profissional.

1.3 Empresa de acolhimento

A Contactus S.A. é uma microempresa de consultoria em sistemas de informação, inserida no mercado das Tecnologias de Informação (TI). Tem como missão o desenvolvimento de serviços, de forma a responder às opções estratégicas dos seus clientes, acrescentado valor aos seus negócios. A sua visão passa pelo seu crescimento, diversificando negócios e lançar-se no mercado internacional, utilizando os seus valores que servem de base ao desenvolvimento do seu projeto empresarial, isto é, ética, ambição, liberdade de opinião, humildade, mérito e confiança.

Liderada e gerida de forma empreendedora, a Contactus S.A. distingue-se pelo seu modelo de organização orientado em processos de negócio, que se dividem em processos primários – produtos e serviços a disponibilizar aos clientes – e processos de suporte ao negócio, que asseguram a atividade da empresa. A Contactus, S.A. é uma empresa ágil e flexível, que se adapta facilmente às constantes evoluções e mudanças dos mercados.

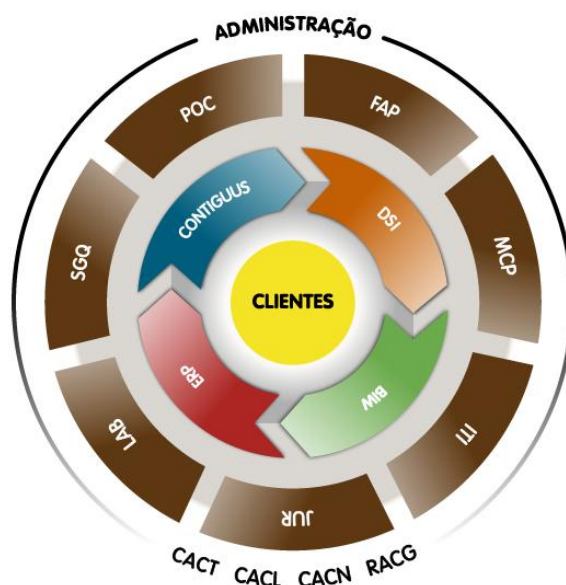


Figura 2 – Modelo Organizacional da Contactus (Fonte: Contactus S.A.)

A Contactus estabeleceu, desde o início, uma parceria com a fornecedora de serviços e *software* TI Oracle, já que esta permitia ter uma grande visibilidade e

notoriedade no mercado, pela sua dimensão e qualidade, podendo a Contactus destacar-se pelo seu conhecimento profundo da tecnologia Oracle, podendo ao mesmo tempo aproveitar vantagens que esta parceria oferece, organizando mais eventos em conjunto com a Oracle, por exemplo. Assim, a Contactus foi reforçando a sua associação estratégica com a Oracle, evoluindo do nível de certificação *Member Partner* para *Certified Partner*, no ano de 2004, chegando em 2008 ao nível máximo de certificação Oracle – *Certified Advantage Partner*. Estas designações sofreram alterações nos últimos tempos, sendo que neste momento, a Contactus detém o nível de parceria *Gold Partner*.

Neste momento, a empresa está dividida em quatro áreas de competência (figura 2), onde estão os colaboradores organizados por áreas, em função das suas competências profissionais, ou seja, Consultores da Área de Competência Técnica (CACT), Lógica (CACL), de Análise de Negócio (CACN) e Recursos da Área de Gestão (RACG). Esta empresa desenvolve a sua atividade em quatro processos primários de negócio, através dos quais implementa, executa o negócio e que gerem valor acrescentado para os clientes. Os processos primários são divididos por *Technical Infrastructures Management* (TIM), *Application Development & Integration* (ADI), *Business Intelligence & Warehouse e Enterprise Resource Planning* (ERP). Com isto, é necessário ter processo de suporte ao negócio que assegurem a atividade da empresa através de suporte aos processos primários de negócio, sendo divididos por Finanças, Administrativo e Pessoal (FAP), Infraestrutura Tecnológica (ITI), Jurídico (JUR), Laboratório de Ideias (LAB), Marketing, Comercial e Pré-Venda (MCP), Planeamento, Organização e Controlo (POC) e Sistema de Garantia de Qualidade (SCQ).

1.4 Integração profissional

No dia 10 de Setembro de 2012 dei início ao estágio nas instalações da Contactus S.A. O primeiro dia na empresa teve como objetivo conhecer a estrutura da empresa, as competências de cada área de negócio, a sua organização e métodos de trabalho. No mesmo dia, tive uma reunião onde me foram apresentados todos os módulos a serem desenvolvidos durante o tempo de estágio e onde me foi facultada toda a informação sobre o projeto.

Durante as primeiras semanas, iniciei um estudo alargado sobre as tecnologias e ferramentas adequadas que teria de utilizar para o desenvolvimento do projeto proposto, tendo mesmo realizado alguns tutoriais para perceber melhor os seus mecanismos.

Posteriormente iniciei o meu projeto de estágio, onde tive o máximo apoio de todos os colaboradores.

1.5 Estrutura do documento

Este documento encontra-se organizado da seguinte forma:

- Capítulo 2 – Especificação de metodologias de desenvolvimento, definição de tarefas e respetiva calendarização;
- Capítulo 3 – Análise e definição da arquitetura da aplicação e definição de requisitos;
- Capítulo 4 – Ferramentas e tecnologias utilizadas ao longo do desenvolvimento do presente projeto;
- Capítulo 5 – Explicação de todo o trabalho realizado ao longo dos nove meses de estágio, detalhando o modelo de dados do sistema e toda a sua correspondente implementação necessária;
- Capítulo 6 – Conclusão – As conclusões finais do projeto, dificuldades encontradas e trabalho futuro;
- Bibliografia e Referências;
- Anexos

Capítulo 2

Metodologia e Planeamento

Neste capítulo, será descrita a metodologia escolhida e utilizada para o desenvolvimento de todo o Projeto de Engenharia de Informática (abordando os motivos que levaram a esta escolha) e a calendarização de todas as fases do projeto, com um resumo sucinto do que é realizado em cada tarefa.

2.1 Metodologia de desenvolvimento

A metodologia utilizada para o desenvolvimento da aplicação foi o modelo em cascata revisto [3]. Esta opção foi tomada, tendo em conta a minha experiência adquirida em pequenos projetos, as limitações de cada metodologia e consequentemente o impacto que teriam na execução deste projeto.

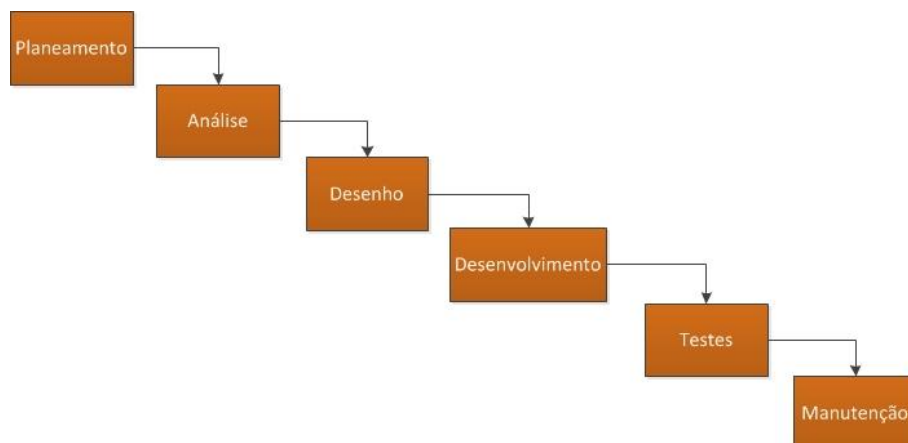


Figura 3 – Modelo em Cascata

O modelo em cascata (figura 3) é um dos processos tradicionais de desenvolvimento de *software*, em que as atividades a executar são agrupadas em tarefas, executadas sequencialmente, de forma que uma tarefa só tenha início quando a anterior tiver terminado. A tarefa seguinte só se inicia quando o cliente valida e aceita o “produto final” da tarefa atual. Quando se avança para a tarefa seguinte, o resultado da tarefa anterior considera-se “fechado” e não pode ser sujeito a posteriores alterações. Este processo de não se poder voltar atrás para alterar modelos e conclusões anteriores, é uma das maiores limitações deste modelo porque, nos tempos que correm, novas ideias estão sempre a surgir para melhoramento do sistema e desta forma não podem ser aproveitadas. É por esta razão que escolhi o “melhoramento” deste modelo.

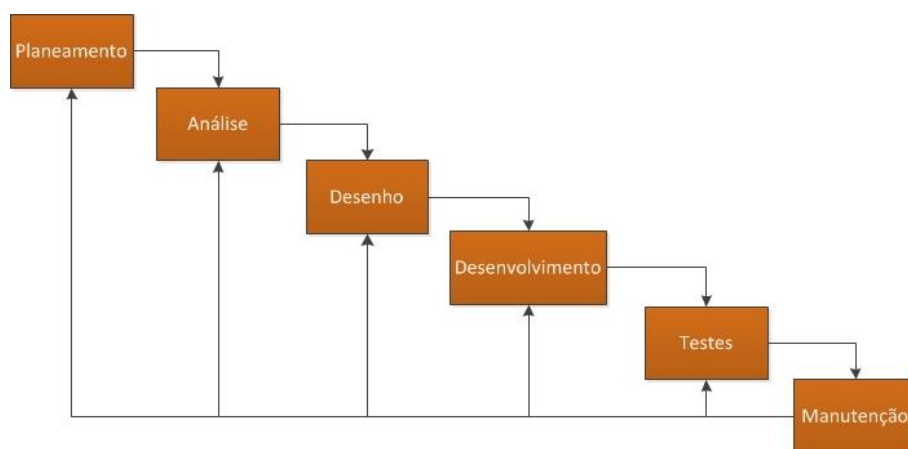


Figura 4 – Modelo em Cascata Revisto

Assim, de forma a eliminar o problema atrás referido foi criado um novo modelo, baseado no modelo clássico em cascata, sendo designado por modelo em cascata revisto (figura 4), que prevê a possibilidade de a partir de qualquer tarefa do ciclo se poder regressar a uma tarefa anterior, de forma a contemplar alterações funcionais e/ou técnicas que entretanto tenham surgido. Claro que este modelo também apresenta limitações, ou seja, na ausência de um processo bem definido de gestão do projeto e de controlo das alterações, podemos passar o tempo num ciclo sem fim, sem nunca se atingir o objetivo final que é disponibilizar um sistema a funcionar.

Tendo em conta que existem outros modelos, como o desenvolvimento iterativo e incremental que se baseia em disponibilizar funcionalidades do sistema mais cedo, e ao longo do tempo ir refinando, alargando pouco a pouco a qualidade, detalhe e âmbito do sistema e o modelo de desenvolvimento ágil de *software* que envolve uma maior interação com o cliente, concordei que o modelo em cascata revisto seria a melhor metodologia a seguir.

2.2 Planeamento do projeto

Na Figura 5, é apresentado o mapa de Gantt [4] relativo ao plano de execução do Projeto de Engenharia Informática decorrido na Contactus, S.A. O trabalho desenvolvido no âmbito do estágio desenrolou-se nas seguintes fases:

- **Fase da Análise** – Nesta etapa deu-se a conhecer com maior detalhe a exposição de todo o projeto a desenvolver, onde houve uma descrição geral do sistema, definição dos objetivos e identificação das ferramentas de desenvolvimento. Na presente fase, criou-se o ambiente de desenvolvimento com a instalação e configuração da base de dados, servidor Apache, PHP e SVN no servidor. Por fim, houve uma clarificação dos requisitos do sistema e do estudo do modelo de dados e das correspondentes funções do sistema.
- **Fase de Conceção / Desenho** – Este período centrou-se em consolidar toda a informação reunida na fase anterior para a realização do desenho da base de dados do sistema e de todos os seus módulos.
- **Fase de Codificação e Testes** – Seguindo a metodologia do modelo em cascata revisto, já referido anteriormente, a presente fase de construção da solução foi efetuada de forma sequencial, isto é, realizou-se a construção da base de dados, de todos os módulos apresentados na figura 5 e os consequentes testes, finalizando-se assim cada etapa, podendo sempre realizar alterações em etapas anteriores.
- **Resultados e Relatórios** – Como fase final do estágio, procedeu-se à recolha e análise de resultados e à elaboração do presente relatório.

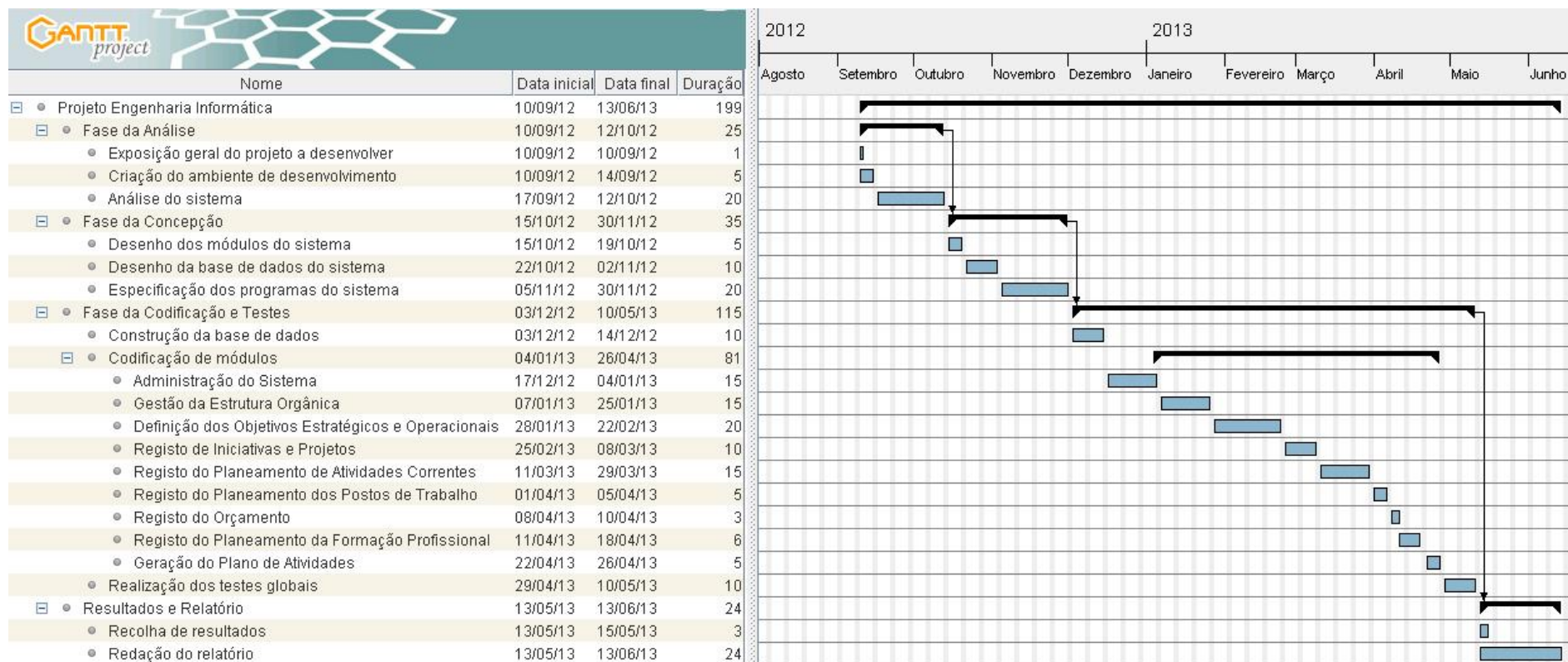


Figura 5 – Calendarização

Capítulo 3

Análise e Desenho

Neste capítulo são apresentados os requisitos definidos pelo cliente, tendo sido analisados e refinados pelos analistas, assim como a arquitetura tecnológica e opções de desenho tomadas para o desenvolvimento do presente projeto.

3.1 Considerações gerais

Antes de mais, é importante uma breve explicação de como é constituída uma Estrutura Orgânica de um Serviço Público, para posteriormente se realizar uma abordagem daquilo que é necessário estar presente num Plano de Atividades.



Figura 6 – Hierarquia da Estrutura Orgânica

Como podemos verificar na figura 6, os Ministérios encontram-se no topo da hierarquia, sendo constituídos por Organismos e estes, por sua vez, são compostos por Unidades Orgânicas, estando estas no fundo da pirâmide. Os Organismos têm sempre uma determinada missão, visão, competências e objetivos que pretendem atingir, mas que podem variar ao longo do tempo. Estes objetivos (estratégicos e operacionais) são o foco principal deste projeto, pois o seu atingimento depende o bom funcionamento do Organismo.

Para o atingimento dos objetivos do Organismo são, anualmente, estipuladas as tarefas de cada Unidade Orgânica e definidos os seus objetivos operacionais parcelares que relacionados com os parâmetros Balanced Scorecard e do Quadro de Avaliação e Responsabilização permitem uma visão da atividade atual da Organização. O controlo de execução e do alcance dos objetivos é efetuado através de indicadores e metas que, periodicamente, informam sobre o avanço e cumprimento dos objetivos inicialmente propostos. A dinâmica da Organização desenvolve-se tendo em consideração a atividade corrente prevista, os projetos a realizar e os recursos humanos e financeiros necessários para a prossecução dos objetivos atrás referidos.

Todos estes elementos e as relações que entre eles se estabelecem e os processos daí decorrentes constituem o Plano de Atividades de uma Organização, cuja informatização é propósito deste projeto.

3.1.1 Balanced Scorecard (BSC)

Como uma ferramenta de Gestão Estratégica surge em 1992, tendo sido apresentado inicialmente como um modelo de avaliação e desempenho empresarial. No entanto, a aplicação em empresas proporcionou o seu desenvolvimento para uma metodologia de gestão estratégica. A partir de uma visão ponderada e integrada de uma organização, o BSC permite descrever a estratégia de uma forma muito clara, segundo novas perspetivas:

- Cliente – Saber qual é o grau de satisfação do cliente;
- Financeiro – Criar novos indicadores de desempenho para que os acionistas possam ter melhor rentabilidade dos seus investimentos;
- Processos Internos – A empresa deve identificar se há produtos com problemas, se foram entregues a tempo previsto e apostar na inovação dos seus produtos;

- Aprendizagem e crescimento (Recursos) – Capacidade e motivação dos trabalhadores, e a um melhor sistema de informação da empresa.



Figura 7 - Hierarquia das Perspetivas do BSC (Fonte: Wikipédia)

O BSC [2] preserva os indicadores financeiros como a síntese final do desempenho gestão organizacional (figura 7), mas incorpora um conjunto de medidas mais genéricas e integradas que vincula o desempenho sob a ótica dos clientes, dos processos internos, dos funcionários e dos sistemas, ao sucesso financeiro de longo prazo.

Basicamente o Balanced Scorecard é um modelo que:

- Disciplina e sistematiza o pensamento estratégico;
- Simplifica o planeamento estratégico;
- Comunica de forma acessível a estratégia;
- Alinha a organização;
- Articula a estratégia com o orçamento e acompanha a execução da estratégia.

É portanto um modelo flexível que deve ser adaptado a cada organização consoante a sua especificidade e dinâmica. O principal objetivo desta metodologia é o alinhamento do planeamento estratégico com as ações operacionais da organização. Esse objetivo é alcançado por esclarecer e traduzir a visão e a estratégia, comunicar e associar objetivos

e medidas estratégicas, planejar, estabelecer metas e alinhar iniciativas estratégicas e melhorar o feedback e aprendizagem estratégica.

3.1.2 Quadro de Avaliação e Responsabilização (QUAR)

A necessidade de um sistema eficaz de gestão operacional e estratégica, toma contornos ainda mais relevantes e atendendo ao Sistema Integrado de Gestão e Avaliação do Desempenho na Administração Pública – SIADAP (aprovado na Lei nº 66-B/2007, de 28 de Dezembro), que no quadro de avaliação de serviços e dos dirigentes, impõe a construção de um QUAR [5] (Quadro de Avaliação e Responsabilização). A avaliação do desempenho de cada serviço assenta, assim, num Quadro de Avaliação e Responsabilização em que se evidenciam os objetivos, indicadores de desempenho, resultados alcançados, meios disponíveis e a avaliação final do desempenho do serviço (figura 8).

No âmbito do Subsistema de Avaliação do Desempenho dos Serviços da Administração Pública (SIADAP 1), o QUAR está sujeito a avaliação permanente e deve ser atualizado a partir dos sistemas de informação do serviço, em que se evidenciam:

- A Missão do serviço;
- Objetivos estratégicos plurianuais determinados superiormente;
- Objetivos anualmente fixados e, em regra, hierarquizados;
- Indicadores de desempenho e respetivas fontes de verificação;
- Os meios disponíveis, sinteticamente referidos;
- Grau de realização de resultados obtidos na prossecução de objetivos;
- Identificação dos desvios e, sinteticamente, as respetivas causas;
- Avaliação final do desempenho do serviço.

Desta forma, o QUAR relaciona-se com o ciclo de gestão do serviço e é fixado e mantido atualizado em articulação com o serviço competente em matéria de planeamento, estratégia e avaliação de cada Ministério. A dinâmica de atualização do QUAR deve sustentar-se na análise da envolvência externa, na identificação das

capacidades instaladas e nas oportunidades de desenvolvimento do serviço, bem como do grau de satisfação dos utilizadores.

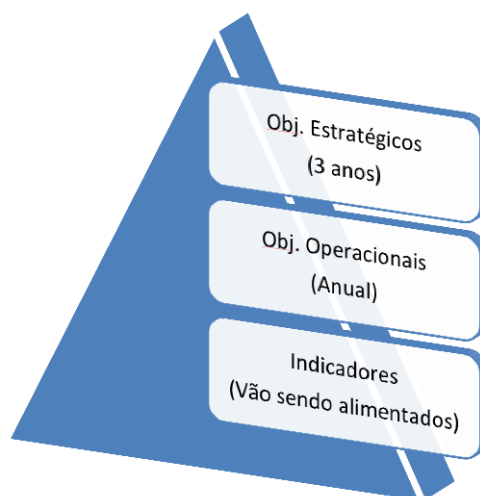


Figura 8 – Hierarquia dos Objetivos de um Organismo

Os serviços iniciam ou prosseguem a construção do QUAR e, no quadro das orientações fixadas pelos respetivos membros do Governo, propõem os objetivos a prosseguir no ano seguinte e estabelecem os indicadores de desempenho e respetivas fontes de verificação.

3.2 Levantamento de Requisitos

Para qualquer projeto que se pretenda desenvolver é necessário realizar o respetivo levantamento de requisitos no sentido de se obter toda informação necessária à formação de uma ideia clara e abrangente daquilo que é necessário fazer, isto é, funcionalidades a desenvolver, dados a utilizar, constrangimentos e restrições. Esta é uma das tarefas de importância fundamental para o sucesso de um projeto, pois a maioria dos projetos fracassados dão-se por erros nos requisitos, sendo estes, provavelmente, a classe de erros mais comuns, originando um consumo de 25% a 40% do orçamento de um projeto.

Relativamente ao Plano de Atividades, os requisitos definidos pelo cliente foram os seguintes:

1. Administração do sistema:

O produto a desenvolver tem de disponibilizar mecanismos que permitam:

- Gestão de utilizadores onde são definidos os utilizadores e as suas permissões de acesso a todo o sistema;
- Definição do nível de acesso aos dados de cada utilizador por diferentes grupos existentes;
- Processos para gerir os menus que estarão disponíveis aos utilizadores.

A gestão de utilizadores é realizada exclusivamente por determinadas pessoas que possuam privilégios próprios para essas funções.

2. Gestão da Estrutura Orgânica

A aplicação deve registar e gerir toda a estrutura orgânica, desde ministérios, organismos e unidades orgânicas, incluindo a missão, visão e competências dos organismos e unidades orgânicas. É fundamental que seja possível criar uma relação hierárquica entre a estrutura orgânica e que seja apresentada graficamente.

3. Gestão de Funcionários

Disponibilização de mecanismos para registar e gerir todos os funcionários (categoria profissional, carreira, posto de trabalho e área de formação) da estrutura orgânica, sendo possível uma visualização gráfica da hierarquia e de toda a estrutura orgânica.

4. Gestão da Estratégia do Organismo baseada na Metodologia *Balanced Scorecard*

Uma das funcionalidades que o sistema terá obrigatoriamente de ter, será a de dar a possibilidade de definir os objetivos estratégicos e operacionais, ou seja, registar e gerir toda a informação relacionada com esses objetivos ao nível do organismo associando-os com os parâmetros BSC e QUAR. Para além dos objetivos ainda deverão ser registados os indicadores e metas a atingir. Mais uma vez, é solicitado que seja possível uma apresentação

gráfica da hierarquia dos objetivos estratégicos e operacionais dos organismos.

5. Gestão das Unidades Orgânicas

Este assemelha-se ao requisito anterior, ou seja, é necessário a definição e gestão de toda a informação relacionada com os objetivos operacionais parcelares, ao nível das unidades orgânicas (direções de serviço), assim como os respetivos indicadores e metas a atingir, sendo também necessário uma visualização gráfica de toda a hierarquia.

6. Planeamento das Atividades Correntes

O sistema deve dispor de mecanismos para gerir toda a informação relacionada com o planeamento de atividades correntes, incluindo os seus destinatários/clientes e afetação de recursos humanos, sendo possível consultar o quadro de resumo de todas as atividades correntes.

7. Planeamento de Projetos

Outro requisito para este projeto, é possibilitar a gestão de toda a informação relacionada com o planeamento dos projetos, incluindo os seus objetivos, indicadores, metas e afetação de recursos humanos. Como podemos verificar, este requisito é muito idêntico ao anterior, sendo necessária também a possibilidade de consultar um resumo da informação sobre os projetos planeados.

8. Gestão dos Postos de Trabalho e Formação Profissional

A aplicação deverá suportar mecanismos para administrar todos os postos de trabalho e ações de formação para os funcionários/unidades orgânicas, sendo apresentado um resumo de toda esta informação que servirá de consulta para qualquer utilizador.

9. Carregar e guardar dados em ficheiro Excel

O produto deve conter mecanismos que possibilitem efetuar o carregamento de informação através de um ficheiro Excel, devidamente formatado. Esta funcionalidade deve estar disponível para carregamento de funcionários, de estruturas orgânicas contendo os seus ministérios, organismos e unidades

orgânicas e os seus objetivos estratégicos e operacionais. Também será possível a funcionalidade inversa, ou seja, guardar informação do sistema para um ficheiro Excel.

10. Gerar o Plano de Atividades

Por fim, e como último requisito, o sistema deverá produzir um ficheiro em PDF relativo à informação registada e gerida no sistema, originando desta forma o documento do Plano de Atividades.

3.3 Arquitetura da aplicação

Nesta secção, pretende-se dar uma vista geral sobre os vários componentes a serem desenvolvidos, a forma como estes vão interagir e padrões de desenho utilizados no projeto. Esta aplicação é constituída por três componentes *Back End*, *Front End* e Base de Dados (SGBD), como podemos verificar na figura 9.

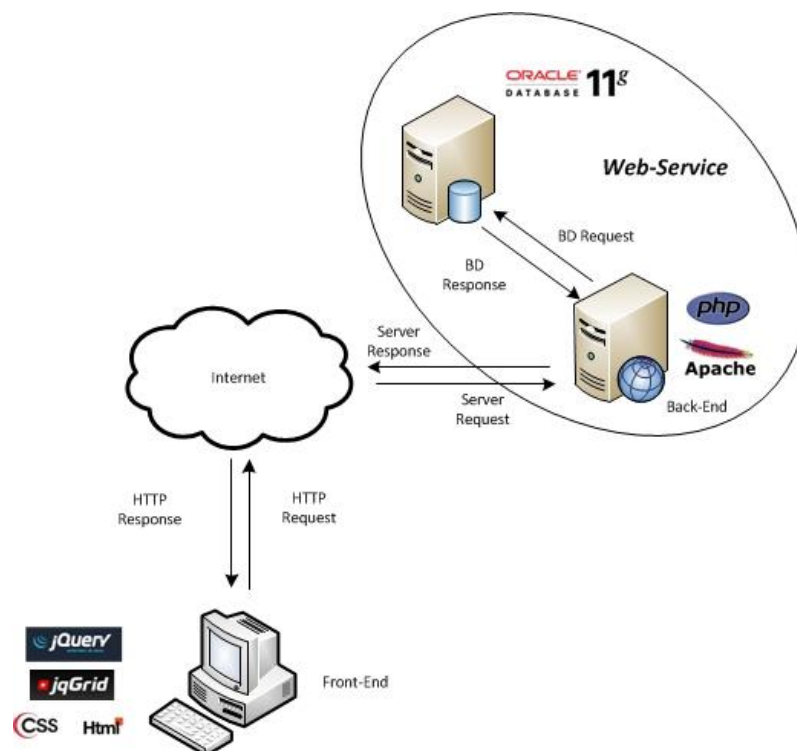


Figura 9 – Arquitetura da aplicação estrategiaGOV

É importante referir que o modelo de arquitetura de *software* que será seguido para o desenvolvimento desta aplicação será o MVC [6] (Model View Controller), como podemos visualizar na figura 10. Esta sugestão foi proposta por mim, perante a empresa, pois com o aumento da complexidade das aplicações desenvolvidas, torna-se fundamental a separação entre os dados (Model) e o *layout* (View). Desta forma, alterações feitas no *layout* não afetam a manipulação de dados, e estes poderão ser reorganizados sem alterar o *layout*. Este modelo resolve o problema através da separação de tarefas de acesso aos dados e lógica de negócio, apresentação e interação com o utilizador, introduzindo um componente entre os dois (Controller).

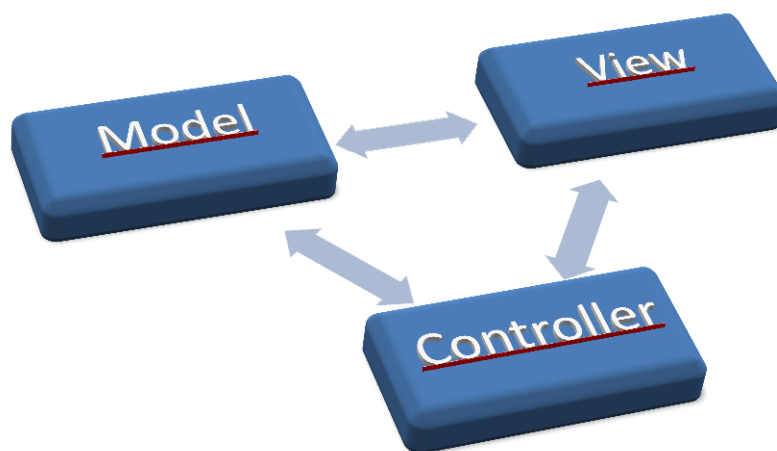


Figura 10 – Modelo MVC (Model View Controller)

Resumindo, o Model é utilizado para manipular informação fornecida pelos serviços do *Back End*, enquanto a View é o responsável pela apresentação do interface ao utilizador, estando localizado no *Front End*. O Controller é responsável por toda a lógica de negócio que é definida no *Back End*, sendo reutilizada pelo *Front End*.

3.3.1 Back End

O *Back End* é a componente onde se situa as tecnologias PHP [7] e Apache [8]. Para a construção de uma aplicação *web* é necessário um servidor *web*, como Apache, sendo este responsável pelo controle de distribuição de páginas na Internet. A função dos servidores *web* é aguardar por pedidos de páginas de clientes provenientes do

browser, e ao receber esse pedido retornar dados ao cliente normalmente em páginas HTML. Estes processos de comunicação entre o cliente e o servidor são feitos por meio do Protocolo de Transferência de Hipertexto [9] (HTTP), que padroniza o modo de envio e recebimentos de dados, tendo como característica principal a sua independência total quanto à plataforma. É aqui que entra a tecnologia Apache, pois é um servidor *web* extremamente configurável, robusto e de alto desempenho, desenvolvido por uma equipa de voluntários (Apache Group), procurando criar um servidor web com muitas características e com código-fonte disponível gratuitamente.

O PHP é uma linguagem que juntamente com o servidor Apache irá responder aos pedidos realizados pelo cliente, sendo que o PHP encarrega-se de realizar processamento de dados enquanto o Apache tem como função o encaminhamento de respostas e pedidos.

3.3.2 Front End

O *Front End* é o responsável por toda a interação direta com os utilizadores através de interfaces gráficas. Este componente tem como função, também, guardar todos os dados fornecidos pelo utilizador, processá-las e adequá-las a uma especificação para o *Back End*. As linguagens aqui utilizadas são jQuery[10], onde serão usadas várias *frameworks*, como é o caso da jqGrid[11], CSS[12] e HTML[13].

3.3.3 Base de Dados

A informação é armazenada em base de dados relacionais, isto é, o Sistema de Gestão de Base de Dados (SGBD) responsável por toda esta componente é Oracle Database 11g. Todo o modelo de dados a ser criado irá ser implementado numa base de dados Oracle [14], visto que oferece uma melhor qualidade de serviço, com poucos riscos de mudança dentro das suas funções e é uma tecnologia bastante completa e robusta. Através do PHP serão realizados pedidos à base de dados, provenientes do utilizador, nos quais este terá de responder, sendo o PHP responsável por interpretar esses resultados. Sendo a Contactus, S.A., Gold Partner da Oracle era inevitável a escolha de outro tipo de base de dados, tendo desta forma maior apoio, caso seja necessário pelos meus colegas de trabalho.

Capítulo 4

Contexto e enquadramento tecnológico

Este capítulo, pretende apresentar o contexto e enquadramento tecnológico em que a aplicação foi desenvolvida, enumerando e descrevendo as tecnologias e ferramentas utilizadas. É importante referir que a maioria das ferramentas e tecnologias aqui descritas foram escolhidas pela empresa, enquanto outras foram sugeridas por mim para um melhor atingimento dos objetivos da aplicação. Inicialmente, serão apresentadas as ferramentas que constituem o ambiente de desenvolvimento utilizado e, de seguida, serão detalhadas as tecnologias escolhidas e como estas se relacionam no sentido de concretizar o sistema proposto.

4.1 Ferramentas de desenvolvimento

Para qualquer projeto que se preze é necessário escolher as melhores ferramentas possíveis, estudando as vantagens e desvantagens das mesmas para que possamos usufruir do máximo das suas funcionalidades. Tendo isto em conta, a lista abaixo descreve as ferramentas utilizadas para a construção do sistema, garantindo a qualidade do *software* desenvolvido.

- **NetBeans** - Disponibiliza um ambiente integrado de desenvolvimento (IDE) totalmente desenvolvida em Java, mas suportando inúmeras linguagens de programação para desenvolvimento, como é o caso do Swing, C, C++, PHP, XML, HTML, JavaScript, CSS, entre outros, sendo um *software open-source*. Este IDE [15] é muito robusto, sendo constituído por uma arquitetura de estrutura reutilizável, permitindo simplicidade no desenvolvimento e aumentando a produtividade. Esta aplicação é constituída por um conjunto

considerável de bibliotecas, módulos, APIs e com toda a documentação necessária possibilitando aos programadores escrever, compilar, realizar “*debug*” das aplicações desenvolvidas de uma forma organizada e estruturada em projetos e módulos. O NetBeans foi uma ferramenta de trabalho que escolhi, tendo em conta que existem outras ferramentas como Eclipse, pois esta é muito mais leve e estável apresentando uma melhor organização de todos os ficheiros e pastas. Outro aspeto a ter em conta, foi que tenho muito maior experiência a trabalhar com este IDE, evitando assim maior perda de tempo na aprendizagem de outra ferramenta.

- **Oracle SQL Developer Data Moduler** - Como o próprio nome indica, é uma ferramenta desenvolvida pela Oracle com o intuito de ajudar na criação dos modelos de dados, através de um excelente interface gráfico. No Data Moduler [16] é possível criar:
 - O modelo lógico dos dados através da definição das entidades, das relações entre entidades, dos atributos e dos domínios;
 - O modelo relacional, deriva do modelo lógico e define tabelas, colunas, chaves primárias e secundários, índices e as relações entre tabelas através das chaves estrangeiras.

Quando o modelo de dados estiver totalmente criado, este cria um Script, com base no modelo relacional com tudo o que foi desenhado para ser executado numa base de dados, dando assim a possibilidade de seleccionar o que é pretendido que seja inserido, ou não, no script.

- **Toad for Oracle** – É um *software* [17] que fornece um conjunto de funcionalidades para desenvolvimento e administração de base de dados Oracle usando comandos SQL. Permite todo o tipo de operações, incluindo adição de colunas a tabelas, restrições, *triggers*, entre outros, mas o que se realça nesta excelente ferramenta é ser extremamente leve, executando *queries* de uma forma bastante rápida e a possibilidade de comparação de dois scripts ilustrando as diferenças entre os dois. Esta ferramenta tem um elevado leque de funcionalidades numa só aplicação evitando a necessidade de utilização de outros *softwares*.

- **SVN** – Serve para gerir arquivos e diretorias armazenados num repositório central ao longo do tempo, ou seja, é um sistema desenhado para controlo de versões. Esta ferramenta [18] é fundamental para programadores, que normalmente passam o tempo fazendo pequenas alterações de *software* e onde existe uma equipa de desenvolvedores a trabalhar em simultâneo sobre o mesmo. O repositório é muito parecido com um servidor de arquivos comum, exceto que este guarda todas as alterações realizadas e por quem foram realizadas, dando a possibilidade de recuperar versões antigas dos arquivos e examinar os dados que foram alterados.

4.2 Tecnologias utilizadas

Neste tópico serão apresentadas as tecnologias utilizadas de forma detalhada para que se entenda o porquê da utilização das mesmas. Para criação de uma aplicação deste nível é inevitável não se utilizar tecnologias como HTML e CSS. Isto porque, HTML é uma linguagem de marcação com objetivo de criar paginas na web, sendo que estas irão ser interpretadas pelos navegadores (browsers), apresentando a correspondente informação, enquanto CSS é uma linguagem de estilo, utilizada para definir a apresentação dessa mesma informação. Com isto, podemos inicializar a apresentação das tecnologias usadas.

4.2.1 jqGrid

Serve para representar e manipular tabelas de dados em aplicações *web*, baseado em Ajax JavaScript (Ajax-enabled JavaScript Control). Isto é, do lado do cliente este carrega os dados de forma dinâmica através de Ajax “*callbacks*”, sendo estes dados obtidos do lado do servidor de forma integrada através de linguagens como PHP, ASP, Java Servlets, JSP, ColdFusion e Perl. É importante referir que a tecnologia Ajax permite uma comunicação assíncrona, através do uso de JavaScript e XML, tornando a aplicação mais dinâmica e interativa com o utilizador.

Projetos						
Organismos do Ministério das Finanças						
	Cód. Organismo	Data Efetiva	Estado	Sigla	Designação curta	Designação longa
▶	92	2012/01/01	Ativo	AT	Autoridade Tributária	Autoridade Tributária e Aduaneira
▶	91	2012/01/01	Ativo	DGT	DGT	Direcção-Geral do Tesouro
▼	90	2012/01/02	Ativo	SGMF	Secretaria-Geral	Secretaria-Geral do Ministério das Finanças
Unidades Orgânicas do Organismo 90						
	Cód. Unid. Org.	Data Efetiva	Estado	Sigla	Designação curta	Designação longa
▶	901	2012/01/02	Ativo	DSIQ	DSIQ	Direcção de Serviços de Inovação e
▶	902	2012/01/02	Ativo	DSGR	DSGR	Direcção de Serviços de Gestão de
▶	903	2012/01/02	Ativo	DSAJC	DSAJC	Direcção de Serviços de Apoio Jurídico
▶	904	2012/01/02	Ativo	DSAD	DSAD	Direcção de Serviços de Arquivos e
▶	905	2012/01/02	Ativo	DSIRP	DSIRP	Direcção de Serviços de Informação
▶	906	2012/01/02	Ativo	UMC	UMC	Unidade Ministerial de Compras

Figura 11 – Gestão de Projetos constituído por Subgrids

Esta ferramenta, além de servir para apresentar a informação, dispõe também de mecanismos para inserir, apagar, visualizar e editar todos os dados inseridos na tabela em tempo real, sendo possível invocar validações já pré-criadas para todos os campos, criar sub-tabelas (*subgrid*), tal como exemplifica a figura 11, fazer pesquisas de forma rápida e eficaz, disponibilizando ao utilizador final um interface bastante atrativo, podendo ser posteriormente modificado pelo utilizador através de um *plugin* do JQuery UI [19].

O jqGrid [11] só consegue importar e exportar dados provenientes em formato XML e JSON, o que faz com que seja obrigatório passar toda a informação entre “etiquetas”, para que sejam interpretadas pelo JavaScript e apresentadas na tabela de forma correta. Caso isso não aconteça nada será apresentado.

4.2.2 Javascript e jQuery

O Javascript é uma linguagem de programação para atender principalmente às necessidades do lado do cliente de uma aplicação *web*. O principal objetivo para o seu desenvolvimento, foi o de criar uma linguagem de script para programação que torne possível a validação de formulários no lado do cliente, através do *browser* e a interação com a página. Uma das principais características do Javascript é o modo como interage com o *browser*, podendo aceder aos seus objetivos.

O jQuery [10] é uma biblioteca Javascript focada na simplicidade, que facilita a adição de efeitos e conteúdos dinâmicos sem a escrita de longas e complexas linhas de código (figura 12).

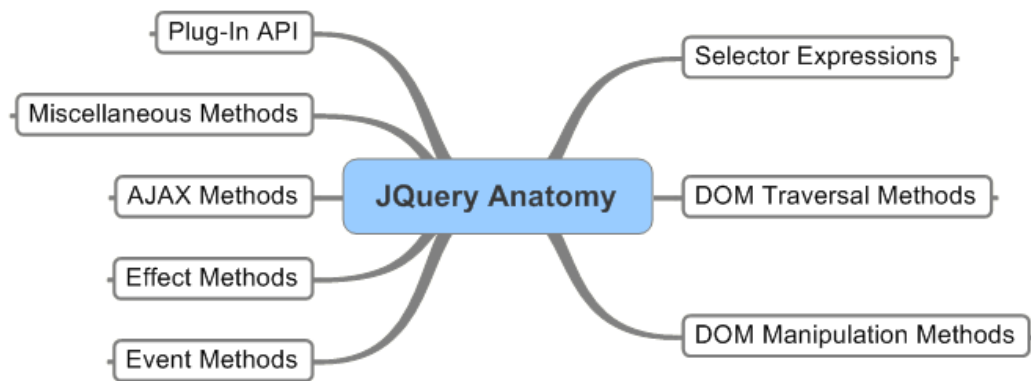


Figura 12 – Arquitetura da Framework jQuery (Fonte: J2eebrain)

Esta tecnologia apresenta várias funcionalidades das quais se destacam:

- Redução de código;
- Resolução da incompatibilidade entre os navegadores;
- Reutilização do código através de *plugins*;
- Utilização de uma vasta quantidade de *plugins* criados por outros desenvolvedores;
- Trabalha com AJAX e DOM;
- Implementação segura de recursos como CSS;

4.2.3 PHP

É uma tecnologia que permite a criação de páginas dinâmicas, que trabalha do lado do servidor, capaz de ser embebido dentro de código HTML e efetuar determinadas operações capazes de gerar páginas instantaneamente [7]. Esta interação pode tornar-se ainda mais interessante quando envolve o acesso à informação armazenada em base de dados e consequentemente a visualização dos dados referentes ao pedido solicitado, suportando mais de vinte servidores de base de dados entre os quais Oracle e MySQL. As principais vantagens do PHP que podemos salientar são:

- Linguagem estruturada e orientada a objetos;
- Portabilidade;
- Velocidade e robustez.

Para testar esta tecnologia é necessário ter instalado o Apache no servidor, isto porque este é compatível com o protocolo HTTP versão 1.1 e as suas funcionalidades são mantidas através de uma estrutura de módulos, sendo possível ao utilizador escrever os seus próprios módulos, recorrendo à API do *software*, ou instalando novas funcionalidades através de *add-ins*, tais como o PHP.

4.2.4 PHPExcel

Possibilita ao utilizador a criação de documentos em Excel, PDF, CSV, HTML, entre outros, baseado em matrizes bidimensionais de PHP de forma simples e leve. Esta potente tecnologia cria uma exportação rápida de uma tabela existente na base de dados para um ficheiro Excel, mas também realiza o processo inverso, ou seja, uma importação. O PHPExcel [20] também dá a possibilidade de mudar determinadas características, como o tipo de letra, a cor do fundo da célula, mudar o nome da “*worksheet*”, fundir várias células, modificar o autor do ficheiro, entre outros, quando guarda informação em ficheiro Excel.

O PHPExcel foi sugerido por mim, pois é uma ferramenta fácil de integrar com a aplicação desenvolvida, disponibilizando a possibilidade de alteração do tipo de ficheiro criado, ou seja, em vez de um ficheiro Excel guardar a informação num PDF. Este aspeto deve ser tomado em conta, isto porque, se refletirmos sobre a possível quantidade de informação guardada num Organismos este poderá exceder as linhas de um ficheiro Excel. Logo, temos duas opções, ou criamos mais que um ficheiro Excel ou então, criamos outro tipo de ficheiro, por exemplo um PDF, onde não existe qualquer limite.

Para finalizar, é importante salientar que este suporta as várias versões de Excel, como 2013, 2010 ou mesmo 2003, fornecendo assim um enorme leque de alternativas ao utilizador.

4.2.5 Oracle

É um sistema de gestão de base de dados relacional (SGBDR), mais fiável e amplamente utilizado. O sistema é construído em torno de uma estrutura de dados relacionais, em que os objetos podem ser acedidos diretamente pelos utilizadores, ou por uma aplicação *front-end*, através de “*Structured Query Language*” (SQL).

Esta tecnologia tem uma arquitetura totalmente escalável e é frequentemente utilizada por empresas globais, que querem processar dados através de redes de longa distancia, ou redes locais, sendo que esta tem um componente próprio de rede que permite essa comunicação. As suas principais características são:

- Gestão de base de dados relacionais com funções bastante complexas;
- Portabilidade (suporta praticamente qualquer plataforma atual);
- Exige *hardware* potente para uma boa performance;
- Necessita de profissionais bastante qualificados para a sua administração;
- Documentação muito bem detalhada, que permite um maior conhecimento deste tipo de base de dados e de todos os seus recursos;
- Controlo de segurança e recuperação de dados.

O Oracle tem um custo mais elevado do que outros SGBD, sendo mais difícil de gerir, porém é um produto que possui mais recursos de segurança e performance, que podem ser muito importantes e cruciais para empresas que possuam aplicações e com grandes volumes de dados que sejam necessários níveis elevados de segurança.

Capítulo 5

Trabalho desenvolvido

O trabalho realizado consistiu na construção do modelo de dados e na implementação de todas as funcionalidades da aplicação de suporte ao Plano de Atividades, no âmbito do sistema estrategiaGOV, abordando as bibliotecas e ferramentas utilizadas no seu desenvolvimento. Para uma melhor exposição das funcionalidades da aplicação, irão as mesmas ser agrupadas em três subconjuntos, como mostra a figura 13.



Figura 13 – Funcionalidades da Aplicação

No grupo 1, e como mostra a figura 14, estão as funcionalidades que permitem ao utilizador definir os elementos necessários para criar um Plano de Atividades, tais como: Estrutura Orgânica (Ministério, Organismo e Unidade Orgânica), Objetivos (Estratégicos e Operacionais), Metodologias escolhidas pelo cliente (Balanced Scorecard e Parâmetros QUAR), Metas a atingir (Indicadores e Metas) e das Atividades Correntes. De relevar também, as funcionalidades referentes à Gestão dos Recursos Humanos e Financeiros e à Administração da aplicação.

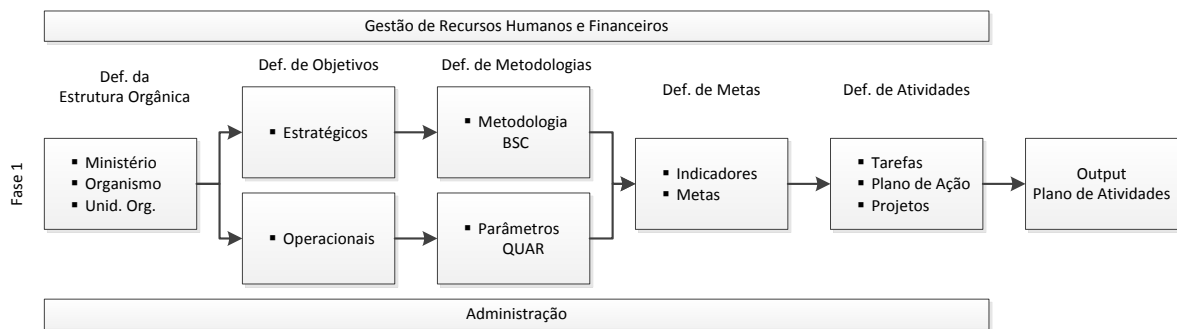


Figura 14 – Definição (Grupo 1)

No grupo 2 (figura 15) é constituída pelo controlo, isto é, serve para carregar e alimentar os indicadores ao longo do tempo, para medir e controlar se os objetivos definidos no início do Plano de Atividades estão a ser cumpridos. Esta fase dispõe também de mecanismos para realizar mitigações para corrigir desvios, caso sejam necessários.

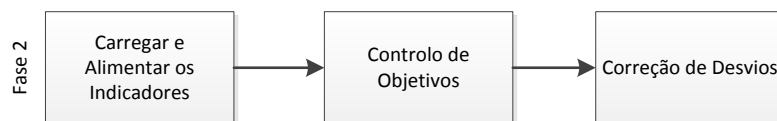


Figura 15 - Controlo (Grupo 2)

Por fim, no grupo 3 (figura 16) irá ser realizado uma análise para verificar se todos os objetivos foram cumpridos como estipulados, analisando sobre a metodologia Balanced Scorecard. Esta metodologia serve de medição e gestão de desempenho entre os objetivos estabelecidos.

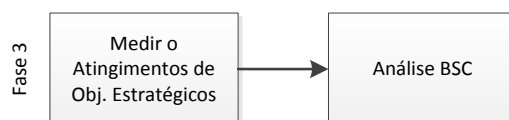


Figura 16 - Medição (Grupo 3)

5.1 Modelo de Dados

Neste tópico será apresentado o modelo de dados criado para implementação de todo o sistema, tendo em conta os requisitos solicitados pelo cliente. Antes de mais, é importante explicar a filosofia de todo este modelo de dados, ou seja, o que se realça neste modelo é a efetividade de todos os dados, aqui armazenados. Desta forma, será possível guardar todo o histórico, tantos dos registos, como das funcionalidades realizadas pelos utilizadores da aplicação, dando assim proteção ao sistema.

Como o modelo de dados é bastante extenso e complexo, irei realizar uma pequena separação para uma mais fácil abordagem, descrevendo, em separado, o modelo de administração do sistema e o modelo do Plano de Atividades, mas claro que ambos se encontram integrados no mesmo modelo.

5.1.1 Administração do Sistema

O presente módulo foi desenhado para realizar a gestão de todos os utilizadores do sistema, estando estes associados a grupos/perfis. Nesta associação é definido a data de início e fim a que o utilizador se encontra associado a esse grupo. Estes grupos estão interligados aos menus existentes na aplicação, para que desta forma se consiga diferenciar quais os menus que os funcionários têm direito a visualizar. Para finalizar esta breve descrição, foi criado uma tabela de acessos que guarda alguma informação dos utilizadores que entrem no sistema.

Como podemos verificar, na figura 18, este modelo é constituído pelas seguintes tabelas:

- **ADM_ACESSOS** – Esta tabela, tem como objetivo guardar dados informacionais de todos os utilizadores que se autenticarem na aplicação, tais como: o seu id de utilizador, IP, o *browser* que está a utilizar e a data do sistema. Desta forma, o administrador de sistema terá sempre um controlo total sobre todos os acessos.

ADM_LOG – Os Log's tem uma importância fundamental em todo o sistema, na medida em que guarda toda a informação, que em determinado modo, tenha sido alterada. Isto é, nesta tabela irá ser registado quem, quando e como é que houve modificações de dados no sistema. O ADM_LOG contém o id do log, o nome da tabela em que foi realizada a alteração, o tipo de operação, ou seja, se foi feito um CREATE, INSERT ou DELETE. A coluna “registro” guarda todos os novos dados que foram inseridos, identificando as colunas que foram objeto dessa alteração. Esta tabela também guarda o IP e o módulo, que serve para informar onde é que foi realizada essa alteração, se diretamente na base de dados ou através de HTTP, como ilustra a figura 17.

Listagem de Histórico						
ID	Tipo de Oper	Utilizador	Data do Sistema	Registro	Alterado em	IP
1	UPD	UZUMAKI	2013/05/22	[DESC_L_MIN=Ministério de Justiça][DESC_C_MIN=Ministério de Justiça]	htpd.exe	192.168.80.106
2	UPD	UZUMAKI	2013/05/22	[DESC_L_MIN=Ministério da Saúde][DESC_C_MIN=Ministério da Saúde][SIGLA_MIN=M]	htpd.exe	192.168.80.106
3	INS	UZUMAKI	2013/05/22	[COD_MIN=4][DATA_EFETIVA=2013/05/01 00:00:00][ESTADO_DTEF=A][DESC_L_MIN=Mir]	htpd.exe	192.168.80.106
11	INS	UZUMAKI	2013/08/05	[COD_MIN=10][DATA_EFETIVA=2012/01/02 00:00:00][ESTADO_DTEF=A][DESC_L_MIN=1]	TOAD background query se:	192.168.80.12
11	DEL	UZUMAKI	2013/08/05	[COD_MIN=][DATA_EFETIVA=][ESTADO_DTEF=][DESC_L_MIN=][DESC_C_MIN=][SIGLA_MIN=]	TOAD background query se:	192.168.80.12

Figura 18 – Listagem de Histórico relativo aos Ministérios

- **ADM_LOG_ATIVO** – Os Log's só serão gerados no caso de estarem registados nesta tabela. O objetivo desta serve para ajudar na performance da base de dados, ou seja, durante o desenvolvimento irão existir variadíssimos dados a ser constantemente inseridos, o que irá originar Log's, então, desta forma, se não estiverem ativos evitar-se-á uma sobrecarga na base de dados, com informação desnecessária.
- **ADM_GRUPOS** – Num sistema desta dimensão é imprescindível a possibilidade da criação de grupos que diferenciem os vários tipos de utilizadores e quais os seus privilégios sobre a aplicação.
- **ADM_GRUPOS_MENUS** – Irá ser utilizada para associar as funcionalidades da aplicação a cada grupo em que o utilizador está inserido. No sistema haverá um mecanismo capaz de associar o ID_GRUPO (ADM_GRUPOS) com o ID_MENU (ADM_MENUS), onde será possível selecionar os serviços aos quais o utilizador terá acesso, como inserir, editar, eliminar ou só mesmo visualizá-los.

- **ADM_MENUS** – Esta tabela foi concebida para registar todos os menus existentes no sistema. A ADM_MENUS é constituída por ID_MENU e ID_MENU_PAI, ou seja, com estes dois campos podemos criar um menu principal, podendo ter vários submenus associados, sendo que para isso é necessário preencher este último campo. Para além disso, nesta tabela é registado o que é visualizado no menu (ETIQUETA), informação sobre a funcionalidade do menu (TIPO_PROGRAMA) e em que pasta é que se encontra a página PHP, seleccionada (PROGRAMA).
- **ADM_UTIL_GRUPOS** – Um pouco à semelhança da tabela ADM_GRUPOS_MENUS, esta associa os grupos aos utilizadores do sistema, sendo possível registar a data de início e fim correspondente.
- **ADM_UTILIZADORES** – Por fim, temos a tabela para registar os dados de utilizadores do sistema, onde se regista toda a informação relacionada com ele, como o número de funcionário, senha, estado da senha, nome e data de criação.

5.1.1 Plano de Atividades

Antes de começar a descrever todas as tabelas existentes neste modelo, é necessário explicar algumas decisões importantes tomadas para que se perceba a complexidade de toda a arquitetura aqui envolvida. Nos tempos que correm, é importante, senão fundamental, guardar todo o histórico de todos os registos alterados ou criados e visto que esta aplicação será utilizada num Organismo, é essencial criar mecanismos para tal. Assim, por um lado, implementou-se a filosofia dos Log's, referidos no módulo anterior, para guardar toda a informação alterada e, por outro, estabeleceu-se efetividade dos dados, definindo para os mesmos a sua própria data de efetividade (DATA_EFETIVA), e não possibilitando a duplicação de registos com a mesma data. Assim, toda a informação alterada ao longo do tempo ficará registada no sistema, podendo ser consultado todo o histórico de cada registo. É também importante referir, que não é possível eliminar informação do sistema, isto é, existe um campo em todas as tabelas designado por ESTADO_DTEF, que indica se a informação correspondente, está ativa (dando continuidade à filosofia de guardar todo o histórico) ou inativa (caso tenha sido eliminado).

Tendo em conta, que o modelo de dados (figura 19) é demasiadamente extenso, só irão ser descritas as tabelas mais importantes relativas ao Plano de Atividades:

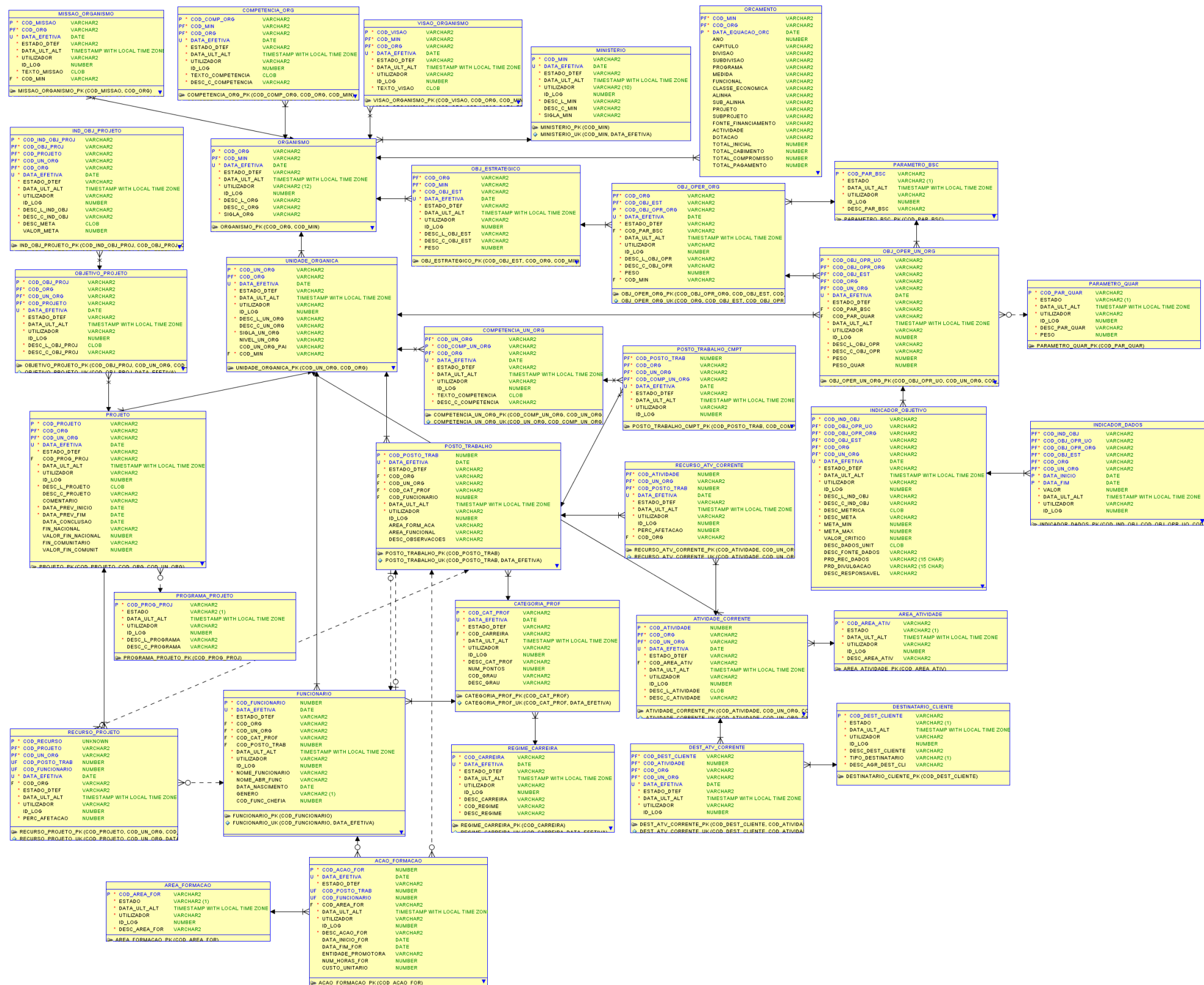


Figura 19 - Modelo de dados relativo ao Plano de Atividades

- **MINISTERIO, ORGANISMO e UNIDADE_ORGANICA** – Servem para registrar toda a informação sobre as Estruturas Orgânicas, estando associadas a outras tabelas para criar as respectivas visões, missões e competências.
- **OBJ ESTRATEGICO, OBJ_OPER_ORG e OBJ_OPER_UN_ORG** – Estas tabelas servem para registrar os objetivos estratégicos e operacionais dos Organismos e das Unidades Orgânicas, sendo estas relacionadas entre si. É importante referir, que para além da informação básica que estas tabelas disponibilizam, existe um campo designado PESO, que serve para registrar a percentagem que um determinado objetivo tem, perante os outros objetivos de um Organismo, nunca podendo ultrapassar, no seu conjunto, os cem por cento. Os Objetivos Operacionais estão relacionados com o parâmetro BSC e com os parâmetros QUAR.
- **INDICADOR_OBJETIVO** – Esta tabela serve para interligar os objetivos estratégicos e operacionais, tanto do Organismo como da Unidade Orgânica delineando as metas, o valor crítico, entre outros.
- **INDICADOR_DADOS** – Esta tabela está associada, a tabela INDICADOR_OBJETIVO, que regista os objetivos que estão a ser cumpridos, servindo desta forma para alimentar os indicadores dos objetivos.
- **FUNCIONARIO** – Aqui é guardado a informação sobre os funcionários existentes para cada Unidade Orgânica, sendo possível criar-se uma hierarquia através do campo COD_FUNC_CHEFIA. Claro que esta tabela liga-se a várias outras, como é o caso do RECURSO_PROJETO, (os recursos humanos necessários à execução de um projeto), CATEGORIA_PROF (corresponde à categoria profissional de cada funcionário), POSTO_TRABALHO (posto de trabalho ao qual o funcionário está afeto) e ACCAO_FORMACAO.
- **ATIVIDADE_CORRENTE** – Esta serve para registrar as atividades correntes de cada Unidade Orgânica, estando separado por áreas de atividade. As atividades correntes também estão ligadas à tabela DEST_ATV_CORRENTE, servindo assim para interligar as atividades aos destinatários/clientes.

- **PROJETO** – Esta tabela guarda toda a informação sobre os projetos a serem realizados pela Unidade Orgânica. Para cada projeto são registados os objetivos a serem cumpridos, os recursos humanos, a afetar as datas previstas de início e fim e os custos financeiros à sua realização.

5.2 Implementação

Nesta secção irão ser descritos todas as funcionalidades implementadas no Plano de Atividades, com auxílio de algumas imagens do interface. Antes da descrição propriamente dita, das funcionalidades vou abordar três aspetos que me parecem importantes para o desenvolvimento do sistema:

- Arquitetura MVC

Esta arquitetura foi implementada tendo em conta os seus respetivos princípios básicos, ou seja, para isto foram criadas três pastas:

- Model - constituído por funções para fazer consultas à base de dados;
- View - que disponibiliza os interfaces mostrados aos utilizadores;
- Controller - constituído pelas funções solicitadas, como adicionar, editar, apagar, entre outros.

- Frameworks

Ponderei em utilizar frameworks de PHP, como o CodeIgniter [21] ou CakePHP [22], mas optei por não o fazer. Com a utilização de uma *framework* iria obter maior facilidade no desenvolvimento da aplicação, mas por outro lado não iria conseguir tanta experiência e conhecimento sobre esta tecnologia.

Todas as páginas solicitadas pelo utilizador serão apresentadas em forma de *grid*, baseadas na ferramenta de trabalho jqGrid, dispondo dos mecanismos básicos como pesquisar, adicionar, editar e remover informação. Estes mecanismos podem ser disponibilizados de duas formas, *inline editing* ou *form editing*. O *inline editing* é uma maneira rápida de atualizar as informações da base de dados, fazendo alterações

diretamente na *grid*, enquanto para *form editing* é apresentada uma janela onde podemos atualizar os respetivos campos. Desta forma, foi delineado que para adicionar, editar e remover iria ser utilizado o *form editing*, enquanto para realizar pesquisas implementou-se o *inline editing*.

- Segurança

Num projeto deste nível, é fundamental criar-se um mecanismo de autenticação assegurando a maior segurança possível, para que seja inacessível a desconhecidos. Assim sendo, criou-se um painel de autenticação para os utilizadores, onde se insere o número do funcionário e senha correspondente com um interface simples. O mecanismo de autenticação é efetuado, através do PHP, com o estabelecimento de ligação ao servidor de base de dados, verificando se este utilizador existe e se a senha coincide com a que está no sistema, sendo esta informação encriptada. Caso os dados inseridos sejam inválidos, será apresentado no *browser* uma mensagem de erro, através de uma função JQuery, para informar o utilizador do motivo do insucesso.

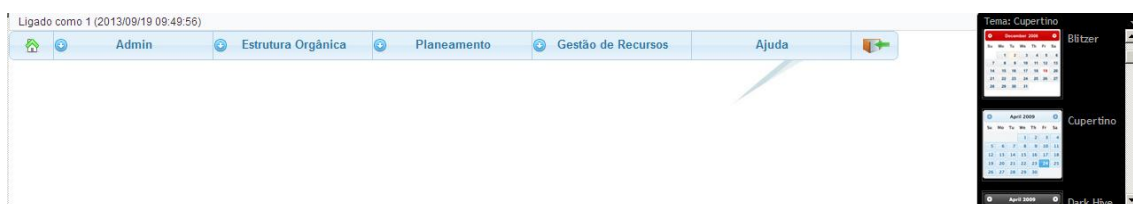


Figura 20 – Menu e ThemeRoller do estrategiaGOV

Após autenticação do utilizador, será exibido um interface com informação do ID do funcionário, data e hora que foi efetuado o login. Os menus que serão apresentados ao utilizador dependem dos privilégios que este detiver no sistema, tendo este sido implementado com ajuda do *plugin* jQuery.ui.potato.menu [23].

- E Parametrização dos ecrãs

É também apresentado uma lista *drop-down*, que foi implementada com ajuda de outro *plugin* jQuery, chamado ThemeSwitcher [24], que dispõe de uma série de temas que o utilizador pode escolher para mudar as cores de *layout*, tanto do menu apresentado como das *grids*, dando assim a possibilidade ao utilizador de escolher o

interface que mais lhe agrada, como podemos verificar na figura 20. De seguida, serão apresentados todos os menus e submenus existentes no sistema.

5.2.1 Administração (Menu Admin)

De seguida, serão apresentados todos os processos necessários para a criação do menu de administração, isto é:

- Gestão de Utilizadores

Na figura 21, podemos visualizar o interface disponível para gestão de utilizadores que se baseia numa *grid*, como referido anteriormente, onde é mostrada a informação de todos os utilizadores existentes na base de dados, como o seu ID, estado da senha, nome, entre outros. A coluna “Permissões” é usada para associar um utilizador a um determinado grupo/perfil, sendo necessário registar as datas de início e fim, através de uma nova janela.





Gestão de Utilizadores									
Utilizador	Estado Senha	Nº Funcionário	Nome	Ativo	Criado em	Criado por	Alterado em	Alterado por	Permissões
5	Ativo	1	Ana Rosa	Inativo	2013/02/04	1	2013/08/07 14:41:12	1	
4	Ativo	4	Pedro Almeida	Ativo	2012/10/12	1	2013/01/04 13:00:36	1	
3	Ativo	3	Ademar Mendes	Ativo	2012/09/28	1	2012/10/17 13:49:56	1	
2	Ativo	2	Raquel Silva	Ativo	2012/09/28	1	2013/08/07 14:41:27	1	
1	Ativo	1	Administrador	Ativo	2012/09/24	1	2012/09/28 11:13:23	2	

Figura 21 – Tabela dos utilizadores

No interface da gestão de utilizadores, no canto inferior esquerdo, podemos verificar a existência de alguns botões que servem para adicionar, editar e eliminar registos, atualizar a informação da tabela e por último a visualização dos Log's por cada registo selecionado. O interface dispõe ainda da possibilidade de listar “x” dados por páginas limitando assim a quantidade de informação mostrada na tabela.

- Gestão de Perfis

Nesta página, são apresentadas três tipos de tabelas (figura 22) onde são exibidos os grupos, os menus e os utilizadores. Ao seleccionar um determinado grupo, serão apresentados todos os utilizadores e menus que pertencem a este mesmo grupo. Na *grid* dos menus é possível gerir quais os privilégios que cada grupo detém, como ler, atualizar, criar e apagar dados.

</

Figura 22 – Painel para administração dos perfis

- Gestão de Menus

Neste submenu é apresentada uma tabela para realizar a gestão de todos os menus e submenus existentes no sistema, definindo todos os campos necessários, como foi referido no modelo de dados.

5.2.2 Estrutura Orgânica

Este processo, e como podemos visualizar na figura 23, consiste no registo da Estrutura Orgânica sendo composto por:

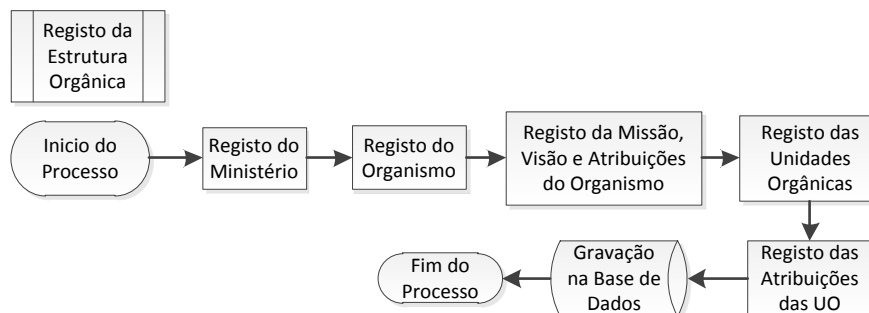


Figura 23 – Processo de Registo da Estrutura Orgânica

- Ministérios

Neste submenu é possível a criação de Ministérios, começando assim por se construir uma Estrutura Orgânica (figura 23). A partir deste submenu, todas as *grids* são constituídas não só pelas funcionalidades básicas, já referidas anteriormente, mas também por outros mecanismos, como o atualizar e corrigir histórico. Quando pretendemos atualizar uma determinada informação, o utilizador é obrigado a mudar a data de efetividade desse registo, alterando alguns campos pretendidos, enquanto no corrigir histórico é apresentada toda a informação guardada e alterada relativa aquele Ministério (figura 24), anteriormente selecionado, proporcionando ainda a correção do histórico guardado.

Ministérios					
Cód. Ministério	Data Efetiva	Ativo	Sigla	Designação curta	Designação longa
9	2013/03/20	Ativo	MF	Ministério das Finanças	Ministério das Finanças
4	2013/05/01	Ativo	ME	Ministério da Economia	Ministério da Economia
2	2013/05/09	Ativo	MJ	Ministério de Justiça	Ministério de Justiça
1	2013/02/05	Ativo	MS	Ministério da Saúde	Ministério da Saúde

Figura 24 – Gestão dos Ministérios

Para além disso, este dispõe de mecanismos para gerar graficamente toda a hierarquia relativa ao ministério selecionado, apresentando desta forma os seus

Organismos e Unidades Orgânicas que o constituem, dando assim uma visão geral da constituição de toda a Estrutura Orgânica. Esta hierarquia gráfica é criada através de um *plugin* jQuery, chamado jOrgChart [25], muito fácil de utilizar, bastando fornecer elementos de HTML para ordenação da informação (figura 25).

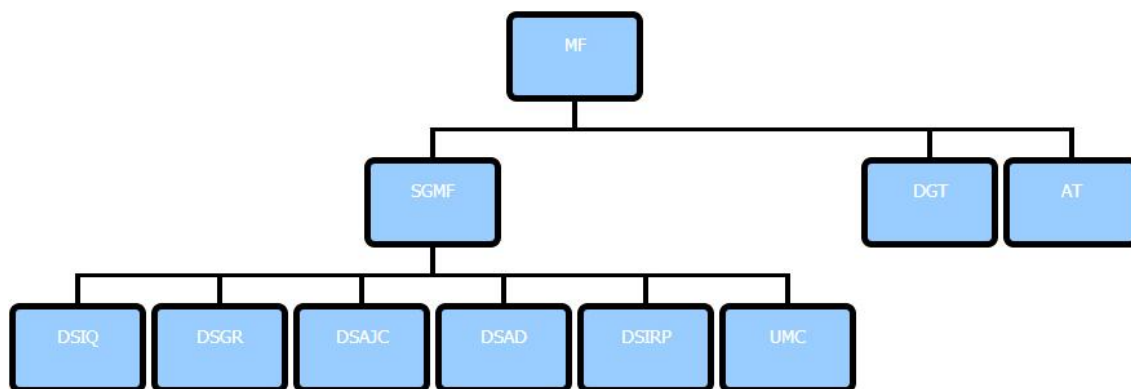


Figura 25 – Hierarquia Organizacional do Ministério

Para além disso, foram ainda desenvolvidos mecanismos para guardar e carregar toda informação através de ficheiros Excel e PDF, sendo que para o carregamento de informação é necessário que o ficheiro esteja corretamente parametrizado, isto utilizando a ferramenta de trabalho PHPExcel. Para finalizar, este ainda contém outros dois botões, um para definir o Ministério como predefinido (pois para visualizar a informação dos outros restantes menus é necessário que o utilizador selecione esta opção) e outro para gerar um ficheiro PDF sobre o plano de atividades do Ministério pretendido.

- Organismos

Esta funcionalidade destina-se à gestão dos Organismos de um determinado Ministério, disponibilizando também algumas funcionalidades semelhantes ao submenu anterior. No entanto, esta *grid* é composta de uma forma diferente, isto porque, para cada Organismo visualizado é possível expandi-lo, observando desta forma três outras *grids* para gerir a sua missão, visão e competências.

Este menu é constituído por outros submenus, ou seja, Objetivos Estratégicos e Operacionais e Parâmetros BSC. A página de Objetivos Estratégicos e Operacionais é constituída mais uma vez, por uma *grid* onde são apresentados todos os Organismos do Ministério predefinido, sendo estas constituídas por *subgrids*, onde são definidos e

geridos os objetivos estratégicos e operacionais do Organismo (figura 26). Esta *grid* dispõe de funcionalidades para guardar e carregar dados através de ficheiros Excel (e PDF) e também consultar todos os objetivos por Organismo, a sua hierarquias e parâmetros BSC e QUAR relacionados.

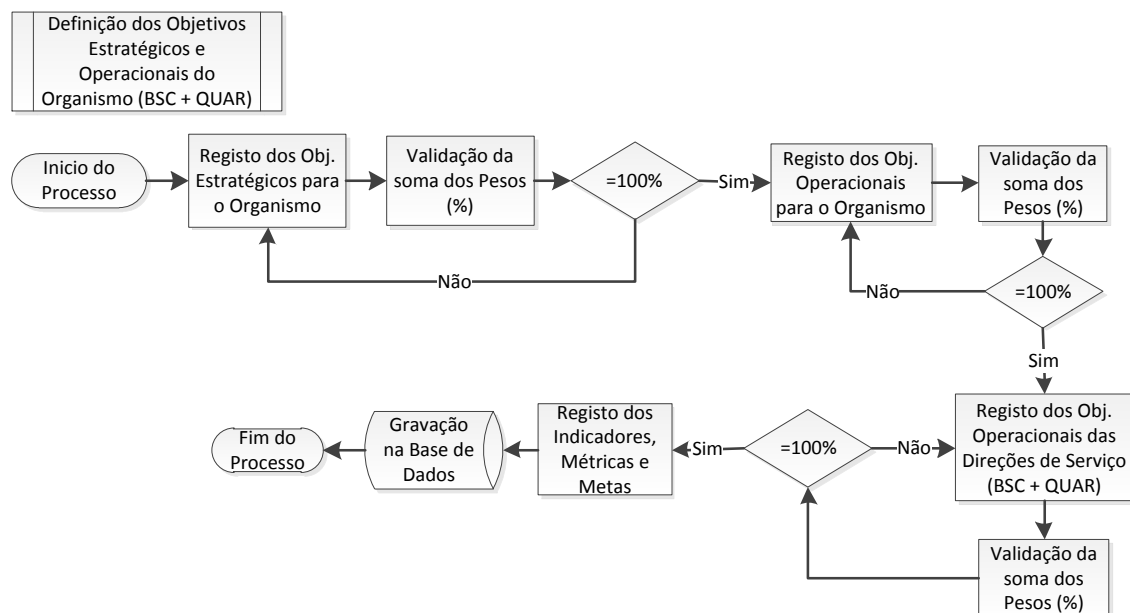
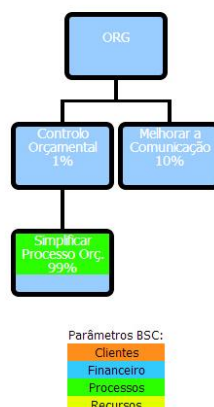


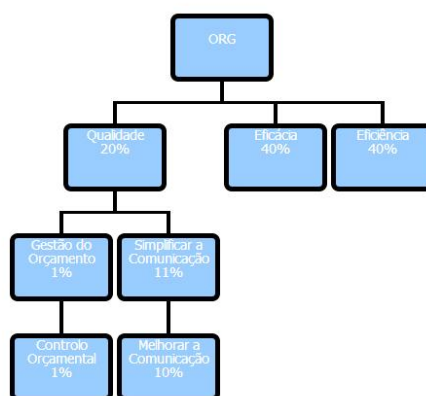
Figura 26 – Processo de definição dos Objetivos Estratégicos e Operacionais (BSC + QUAR)

Ao observar a figura 27, podemos visualizar três tipos de hierarquias. Na primeira, verificamos que começa com o Organismo previamente selecionado, com os seus objetivos estratégicos e objetivos operacionais associados em perspetiva com os parâmetros BSC correspondentes. Na hierarquia seguinte, estão destacados os parâmetros QUAR (de eficiência, de eficácia e de qualidade) que o Organismo se propôs cumprir. Cada um destes parâmetros está associado aos objetivos operacionais que foram delineados no sistema, sendo estes agrupados por objetivo estratégico. Por último, temos um diagrama que mostra os objetivos estratégicos suportados pelos respetivos objetivos operacionais, sendo estes associados aos parâmetros QUAR. As percentagens visualizadas nos diagramas dizem respeito ao peso de cada um dos objetivos ou dos parâmetros QUAR, em que a soma não pode ultrapassar os cem por cento.

Objectivos Estratégicos e Operacionais por parâmetros BSC



Objectivos Operacionais por parâmetro QUAR e por Objectivo Estratégico



Objectivos Operacionais por Objectivo Estratégico

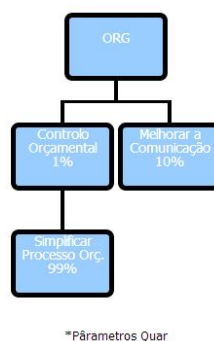


Figura 27 – Hierarquia dos objetivos estratégicos e operacionais

- Unidades Orgânicas

Nesta página é apresentada ao utilizador uma “grelha” com os Organismos existentes e por uma *subgrid* constituída pelas Unidades Orgânicas, sendo esta, por sua vez, constituída por outra *subgrid* para delinear as suas competências. Um pouco à

semelhança do menu “Organismo”, este também é constituído por submenus: Objetivos Operacionais das Unidades Orgânicas e Parâmetros QUAR.



Figura 28 – Gráfico gerado para Indicadores de um determinado objetivo

É importante explicar as funcionalidades existentes no submenu Objetivos Operacionais das Unidades Orgânicas, pois este é composto por uma série de *subgrids* para gerir os objetivos operacionais das Unidades Orgânicas, os seus indicadores e os dados necessários para alimentar esses indicadores. Associados aos Objetivos, estão indicadores que, em modo gráfico informam se aqueles objetivos estão a ser ou não, cumpridos. Como podemos verificar na figura 28, este gráfico apresenta três cores diferentes, ou seja, caso o apontador estivesse situado na zona vermelha, significava que os objetivos mínimos ainda não tinham sido alcançados. A zona verde revela que os objetivos mínimos já foram cumpridos mas que ainda não atingiram o máximo delimitado, enquanto a zona amarela mostra que objetivos máximos já foram ultrapassados.

5.2.3 Planeamento

- Atividades correntes

Esta funcionalidade suporta a gestão das atividades correntes de cada Unidade Orgânica, como apresentado na figura 29. Nesta *grid*, o utilizador dispõe de um botão para poder visualizar todas as atividades correntes relativas a um determinado

Organismo, previamente selecionado, com informação relativa às atribuições e processos por cada área de atividade.

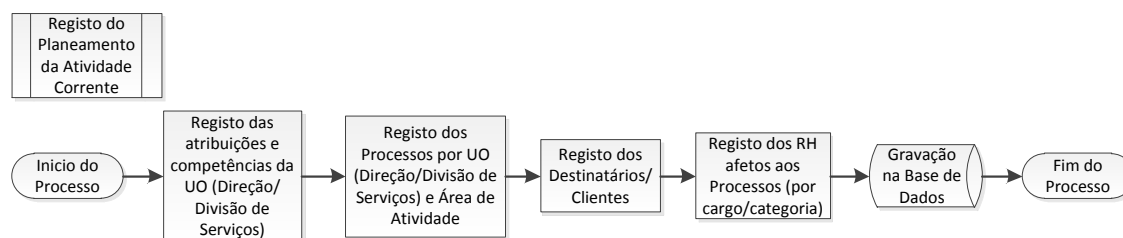


Figura 29 - Processo do Planeamento de Atividades

Estes encontram-se divididos por Unidade Orgânica, apresentando desta forma, um resumo de toda a informação sobre as atividades correntes (figura 30). Esta tabela é composta por uma *subgrid*, onde é possível definir os destinatários de cada atividade corrente.

Resumo da informação sobre as atividades correntes					
Unidade Ministerial de Compras (UMC)					
SÍNTESE DAS ATRIBUIÇÕES E COMPETÊNCIAS DA UNIDADE ORGÂNICA					
Adquirir novos colaboradores Assegurar as funções de Unidade Ministerial de Compras					
PROCESSOS - CARACTERIZAÇÃO SUMÁRIA Quantificação/Destinatários					
Area de Atividade	Processo nº	Listagem e descrição dos principais processos	Destinatário		
			Externos	Internos	
A1	1	Processo 1	X		
OPERACIONALIZAÇÃO DAS COMPETÊNCIAS DA UNIDADE ORGÂNICA PROCESSOS DESENVOLVIDOS POR ÁREAS DE ACTIVIDADE					
Processos		Identificação dos Destinatários ("clientes")			
		Clientes Externos	Nº	Clientes Internos	Nº
Nº de processos internos		0			0

Figura 30 – Resumo das Atividades Correntes de um Organismo

- Projetos

Esta funcionalidade suporta a gestão dos projetos delineados por cada Unidade Orgânica, definindo desta forma, o programa de enquadramento, os seus objetivos, indicadores e recursos associados (figura 31).

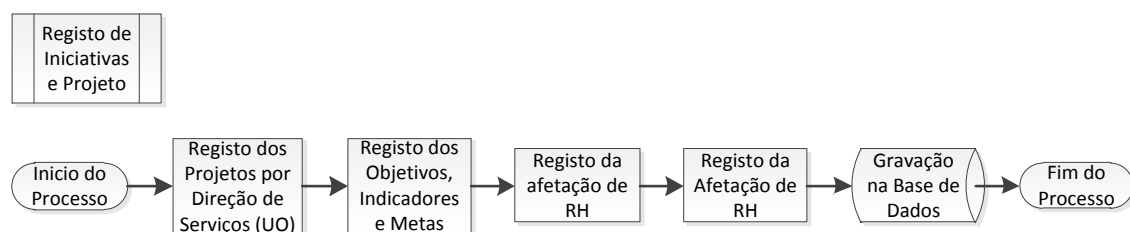


Figura 31 - Processo de Registo de Iniciativas e Projetos

Outra das funcionalidades aqui disponíveis aos utilizadores é que estes dispõem de um gráfico com o resumo de todos os projetos existentes relativos a cada Unidade Orgânica (figura 32), focando-se assim na designação, objetivos, indicadores e data da previsão de conclusão.

UMC				
Designação		Objetivos	Indicadores	Metas
Programa 1	Projeto 1 2013/04/01-2013/05/30	-Melhorar os Sistemas de Inf.	-Indicador 1	-Meta 1

Figura 32 – Resumo dos projetos de um Organismo

Aqui também é possível afetar os recursos humanos necessários, com base nos funcionários geridos no menu Gestão de Recursos.

5.3 Gestão de Recursos

- Funcionário

Nesta página é realizada a administração dos funcionários (figura 33) relativos a cada Estrutura Orgânica criada no sistema. Este assemelha-se ao submenu Ministérios, visto ser constituído por funções semelhantes, como carregar e guardar informação dos funcionários em ficheiro Excel (e PDF) e também consultar a sua hierarquia. Esta hierarquia apresenta o Ministério, dividido em Organismos, que por sua vez se dividem nas respetivas Unidades Orgânicas.



Figura 33 – Processo de Planeamento dos Postos de Trabalho (RH)

O próximo nível representa os diferentes funcionários registados por cada Unidade Orgânica, sendo que os funcionários ficam situados num primeiro nível e caso sejam funcionários chefes irão conter um segundo nível.

Este é constituído por vários submenus relativos à categoria profissional, regime, posto de trabalho e ação de formação de cada funcionário, como apresenta a figura 34.

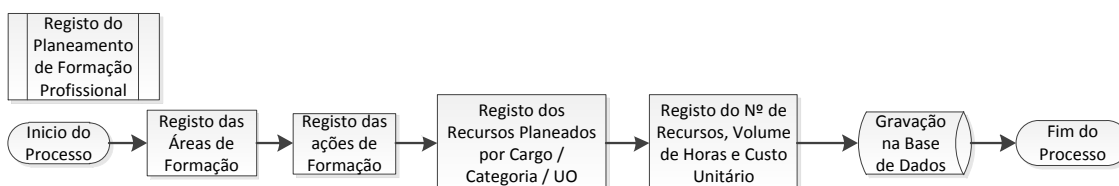


Figura 34 - Processo de Planeamento da Formação Profissional

5.3.1 Ajuda

Este “ecrã” tem como objetivo explicar as funcionalidades de cada um dos menus, para que desta forma, todos os utilizadores fiquem com uma noção do que cada um destes faz, apesar de haver sempre uma formação aos utilizadores que tenham de utilizar este sistema. Este menu pode parecer desnecessário à primeira vista mas é muito

importante, e é muitas vezes esquecido em aplicações. Os responsáveis pelo planeamento de atividades, e por toda esta gestão, poderão ser substituídos ao longo do tempo o que faz com que a funcionalidade “Ajuda” seja bastante relevante quando os utilizadores não saibam ou não se lembrem quais as funcionalidades que este sistema disponibiliza.

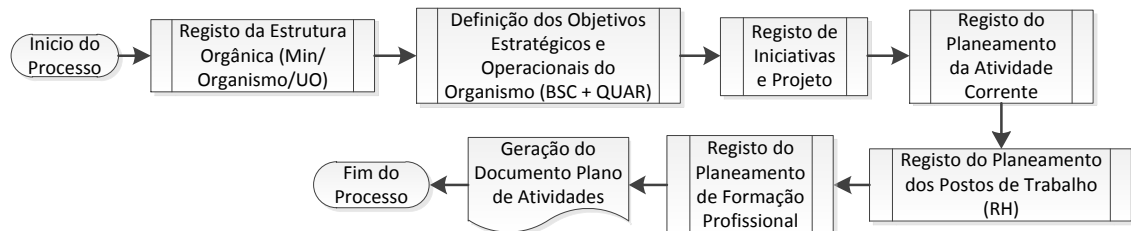


Figura 35 – Visão geral da construção do Plano de Atividades do Organismo

Na figura 35, podemos verificar a junção de todos os módulos referidos anteriormente, dando uma visão geral da complexidade de todo o trabalho aqui desenvolvido.

Capítulo 6

Conclusão

Neste capítulo será apresentado, o resumo do trabalho realizado, o trabalho a ser concretizado futuramente e um breve comentário crítico de todo o projeto desenvolvido.

6.1 Trabalho realizado

O presente relatório apresentou todas as fases que constituíram o desenvolvimento da aplicação *web* de suporte ao Plano de Atividades de um Organismo, no âmbito do sistema *strategiaGOV*. Todos os objetivos foram cumpridos e todos os requisitos do cliente foram implementados com êxito, originando assim uma aplicação robusta, segura e com interface bastante amigável para o utilizador, sendo suportado pela maioria dos *browsers*.

Esta aplicação *web* é constituída por diversos mecanismos e funcionalidades que facilitam todo o processo de gestão do plano de atividades. As vantagens principais que este sistema realça são a possibilidade de administração de utilizadores guardando o histórico de todas as alterações realizadas por estes, gerir toda a Estrutura Orgânica e os correspondentes objetivos operacionais e estratégicos e planear todos os projetos, gerindo assim os correspondentes recursos humanos. É também de realçar, a possibilidade de carregar e guardar informação em ficheiros Excel (e PDF) e visualização gráfica da organização da Estrutura Orgânica, objetivos operacionais e estratégicos, projetos e atividades correntes.

Deste modo, podemos afirmar que esta aplicação aqui desenvolvida está preparada para ser instalada em qualquer Organismo que pretenda um sistema para tais funções. Todas as funcionalidades necessárias para a criação do Plano de Atividades foram implementadas, contendo também mecanismos para gerir os objetivos que foram

delineados, tendo-se criado uma aplicação bastante robusta e com interface bastante apelativa para quem a utilize.

Com isto, conclui o desenvolvimento do Projeto e consequentemente o estágio na empresa Contactus S.A., onde obtive maiores conhecimentos sobre novas tecnologias e conceitos tornando-se assim uma experiência muito importante para trabalhos futuros e também contribuindo para a minha experiência profissional.

6.2 Trabalho Futuro

Neste momento, esta aplicação encontra-se finalizada estando pronta para entrar em produção. Contudo, em termos de trabalho futuro, continuarão sempre a existir, pelo menos, tarefas de apoio à produção e alguns melhoramentos. Um dos aspetos que devia ser melhorado era a apresentação das hierarquias dos objetivos, isto porque, para além do tamanho das “caixas” serem demasiado pequenas para objetivos com descrições mais extensas, deve-se criar outro tipo de mecanismos para diferenciar os parâmetros Balanced Scorecard e do Quadro de Avaliação e Responsabilização.

Outro melhoramento a ser realizado, é o aperfeiçoamento do menu para gerir objetivos operacionais das unidades Orgânicas, pois este é constituído por demasiadas *subgrids*, o que poderá tornar-se confuso para os utilizadores.

Por fim, e não menos importante, deveriam ser criados mecanismos para que os indicadores fossem alimentados de forma automática, consoante os objetivos cumpridos, e não de forma manual.

6.3 Comentário Crítico

Do meu ponto de vista, este projeto era demasiado extenso para um período de desenvolvimento de nove meses, não só, por desconhecer a maioria das tecnologias que utilizei ao longo do projeto, mas também, e como podemos verificar, o modelo de base de dados é muito elaborado e demorou bastante tempo até chegar a esta versão final. Mas provavelmente se a exigência requerida tivesse sido menor, não teria aprendido metade das coisas que aprendi.

As tecnologias utilizadas não precisam de qualquer tipo de licenciamento, excepto a base de dados Oracle, podendo ser gratuita, dependendo do tipo de produto escolhido

pelo cliente. Este ponto é bastante importante, pois, é necessário ter em conta que atualmente os clientes pretendem aplicações com o mínimo de licenciamento possível para evitar despesas.

Um dos aspetos mais positivos no projeto, na minha opinião, é a experiência que obtemos quando estamos a desenvolver aplicações para um cliente. Pois, temos de ter em atenção todos os conhecimentos adquiridos ao longo do percurso académico, e também ter em conta que as aplicações têm de ser desenvolvidas com o máximo de rigor possível. Ao longo do desenvolvimento da aplicação obtive apoio de diversas pessoas da empresa, para além do meu orientador, o que me facilitou uma adaptação bastante fácil, o que cria um bom ambiente de trabalho incentivando assim o gosto pelo trabalho desenvolvido.

Logo, no geral, faço um balanço muito positivo relativamente ao projeto e à minha estadia na empresa Contactus, S.A.

Bibliografia e Referências

[1] CONTACTUS S.A. – **Contactus S.A.** [Em linha]. Contactus S.A. [Consult. 6 Julho 2012]. Disponível na WWW:<URL: <http://www.contactus.pt>>.

[2] WIKIPEDIA – **Balanced Scorecard** [Em linha]. Wikipédia [Consult. 21 Set. 2012]. Disponível na WWW:<URL: http://en.wikipedia.org/wiki/Balanced_scorecard>.

[3] **VIDEIRA**, Carlos; **SILVA**, Alberto – *UML, Metodologias e Ferramentas Case – Vol. I*. 2ª edição. Centro Atlântico 2005

[4] GANTT PROJECT – **GanttProject** [Em linha]. GanttProject [Consult. 1 Nov. 2012]. Disponível na WWW: <URL: <http://www.ganttproject.biz>>.

[5] CGA – **Quadro de Avaliação e Responsabilização** [Em linha]. Caixa Geral de Aposentações [Consult. 21 Set. 2012]. Disponível na WWW:<URL: <http://www.cga.pt/quadroavaliacao.asp>>.

[6] WIKIPEDIA – **Model-View-Controller** [Em linha]. Wikipédia [Consult. 10 Set. 2012]. Disponível na WWW:<URL: <http://pt.wikipedia.org/wiki/MVC>>.

[7] **SERRÃO**, Carlos; **MARQUES**, Joaquim – *Programação com PHP 5.3* – FCA – Editora Informática 2009

[8] APACHE – **Apache Foundation** [Em linha]. Apache [Consult. 11 Set. 2012]. Disponível na WWW:<URL: <http://www.apache.org>>.

[9] WIKIPEDIA – **Hypertext Transfer Protocol** [Em linha]. Wikipédia [Consult. 15 Maio 2013]. Disponível na WWW:<URL: http://pt.wikipedia.org/wiki/Hypertext_Transfer_Protocol>.

[10] JQUERY – **JQuery** [Em linha]. JQuery [Consult. 12 Set. 2013]. Disponível na WWW:<URL: <http://jquery.com>>.

[11] TRIRAND – **jqGrid** [Em linha]. Trirand [Consult. 12 Set. 2013]. Disponível na WWW:<URL:<http://www.trirand.com/blog>>.

[12] WIKIPEDIA – **Cascading Style Sheets** [Em linha]. Wikipedia [Consult. 13 Set. 2013]. Disponível na WWW:<URL: http://pt.wikipedia.org/wiki/Cascading_Style_Sheets>.

[13] HTML – **HyperText Markup Language** [Em linha]. HTML [Consult. 15 Maio 2013].

Disponível na WWW:<URL: <http://pt.wikipedia.org/wiki/HTML>>.

[14] ORACLE – **Oracle Database** [Em linha]. Oracle [Consult. 10 Set. 2012].

Disponível na

WWW:<<http://www.oracle.com/br/products/database/overview/index.html>>.

[15] NETBEANS – **Netbeans IDE** [Em linha]. Netbeans [Consult. 10 Set. 2012].

Disponível na WWW:<<http://www.netbeans.org>>.

[16] ORACLE – **Oracle SQL Developer Data Modeler** [Em linha]. ORACLE

[Consult. 22 Set. 2012]. Disponível na

WWW:<<http://www.oracle.com/technetwork/developer-tools/datamodeler>>.

[17] QUEST - **Toad for Oracle** [Em linha]. Quest [Consult. 12 Set. 2012]. Disponível na WWW:<<http://www.quest.com/toad-for-oracle>>.

[18] WIKIPEDIA – **SVN** [Em linha]. Wikipédia [Consult. 1 Out. 2012]. Disponível na WWW:<<http://pt.wikipedia.org/wiki/Subversion>>.

[19] JQUERYUI - **jQuery UI** [Em linha]. JQuery UI [Consult. 22 Set. 2012].

Disponível na WWW:<<http://jqueryui.com>>.

[20] PHPEXCEL – **PHPEExcel** [Em linha]. PHPEExcel [Consult. 22 Out. 2012]. Disponível na WWW:<<http://phpexcel.codeplex.com>>.

[21] ELLISLAB – **CodeIgniter** [Em linha]. EllisLab [Consult. 12 Set. 2012]. Disponível na WWW:<<http://ellislab.com/codeigniter>>.

[22] CAKEPHP – **CakePHP** [Em linha]. CakePHP [Consult. 12 Set. 2012]. Disponível na WWW:<<http://cakephp.org>>.

[23] MAKOTOKW – **jQuery.ui.potato.menu** [Em linha]. Makotokw [Consul. 12 Set. 2012]. Disponível na WWW:<http://makotokw.github.io/jquery/ui_potato_menu>.

[24] GITHUB – **ThemeSwitcher** [Em linha]. Github [Consult. 12 Set. 2012]. Disponível na WWW:<<https://github.com/harborhoffer/Super-Theme-Switcher>>.

[25] GITHUB – **jOrgChart** [Em linha]. Github [Consult. 12 Set. 2012]. Disponível na WWW:<<https://github.com/wesnolte/jOrgChart>>.